ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

ÖNERİ SİSTEMLERİNDE BAŞARIMI ARTIRMAK İÇİN YAPAY ZEKA TABANLI YAKLAŞIMLAR

Berna ŞEREF

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2021

ÖZET

Doktora Tezi

ÖNERİ SİSTEMLERİNDE BAŞARIMI ARTIRMAK İÇİN YAPAY ZEKA TABANLI YAKLAŞIMLAR

Berna ŞEREF

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Gazi Erkan BOSTANCI

Yapay zeka, insan zekasını ve davranışlarını öğrenme ve taklit etme üzerine kurulu bir alandır ve yapay zeka teknikleri akıllı öneri sitemlerinde, akıllı karar destek sitemlerinde, oyun programlama, robotik, örüntü tanımlama gibi birçok işlevlerde kullanılmaktadır. Teknolojinin gelişmesiyle, bilgi aşırı derecede artmış, sonuç olarak yararlı bilgiye erişim ve bilginin yorumlanıp değerlendirilmesi zorlaşmıştır. Bu zorluğu aşmak adına kullanılan akıllı öneri sistemleri bireylerin duygu ve düşüncelerini de hesaba katarak onlara kısa sürede ve kişiye özel önerilerde bulunabilmektedir. Akıllı öneri sistemlerinin kullanıldığı alanların başında e-ticaret siteleri, haber ve makale siteleri gelmektedir. Bu tez çalışmasında yapay zeka tabanlı yaklaşımlar ile öneri sistemlerinin başarısının artırımı amaçlanmıştır. İlk aşamada, Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının performansı karşılaştırılmış, bu iki algoritma için en iyi doğruluk performansını gösteren yumuşatma, normalizasyon ve ağırlık parametreleri tespit edilmiştir. Tespit edilen parametreler kullanılarak bu iki algoritmayla duygu analizinde bulunulmuştur. Ardından, kullanıcı tabanlı bir öneri sistemi geliştirilmiş, kullanılan benzerlik hesaplama yöntemlerinin tahmin performansına olan etkisi gözlenmiştir. Son olarak, yapay sinir ağları kullanılarak öneri sisteminin tahmin performansı iyileştirilmeye çalışılmış, ardından, yapay sinir ağlarına genetik algoritma uygulanmıştır. Önerilen sistemin performansının literatürdeki güncel ve öncü çalışmaların performansından daha iyi olduğu görülmüştür.

Haziran 2021, 108 sayfa

Anahtar kelimeler: Yapay zeka, öneri sistemleri, yapay sinir ağları, genetik algoritma, tahmin, sınıflama, büyük veri, büyük veri problemleri

ABSTRACT

Ph.D. Thesis

ARTIFICIAL INTELLIGENCE BASED APPROACHES TO INCREASE SUCCESS IN RECOMMENDER SYSTEMS

Berna ŞEREF

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Gazi Erkan BOSTANCI

Artificial intelligence is a field based on learning and imitating human intelligence and behavior, and artificial intelligence techniques are used in many functions such as smart recommender systems, smart decision support systems, game programming, robotics, pattern identification. With the advancement of technology, information has grown enormously, as a result, access to useful information, interpretation and evaluation of it has become difficult. Intelligent recommender systems, which are used to overcome this difficulty, take into account the feelings and thoughts of individuals and can make personalized recommendation to them in a short time. The main areas where smart recommender systems are used are e-commerce, news and article sites. In this thesis, it is aimed to increase the success of recommender systems with artificial intelligence-based approaches. In the first stage, the performance of Naive Bayes and Complementary Naive Bayes algorithms was compared, and the smoothing, normalization and weight parameters showing the best accuracy performance for these two algorithms were determined. Emotion analysis was performed with these two algorithms using the determined parameters. Then, a user-based recommendation system was developed and the effect of the similarity calculation methods used on the estimation performance was observed. Finally, by using artificial neural networks, the predictive performance of the recommender system has been tried to be improved, then a genetic algorithm has been applied to artificial neural networks. It has been observed that the performance of the proposed system is better than the performance of current and pioneering studies in the literature.

June 2021, 108 pages

Key Words: Artificial intelligence, recommender systems, artificial neural networks, genetic algorithm, prediction, classification, big data, big data problems

TEŞEKKÜR

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyerek akademik ortamda olduğu kadar beşeri ilişkilerde de engin fikirleriyle yetişme ve gelişmeme katkıda bulunan danışman hocam Sayın Doç. Dr. Gazi Erkan BOSTANCI'ya, çalışmalarım süresince maddi manevi desteklerini esirgemeyen Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı öğretim üyelerinden Sayın Doç. Dr. Recep ERYİĞİT'e, çalışmalarım sırasında önemli katkılarda bulunan ve yönlendiren Sayın Yrd. Doç. Dr. Ahmet Ercan TOPCU'ya (Ankara Yıldırım Beyazıt Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı Öğretim Üyesi), bugüne kadar gösterdikleri eşsiz sevgi, destek, fedakarlık ve hoşgörü nedeniyle değerli aileme teşekkürlerimi sunarım.

Berna ŞEREF
Ankara, Haziran 2021

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

Büyük veri terimi, gelişen teknoloji sonucunda oluşan veri miktarının artışıyla birlikte günümüzde en çok kullanılan terimlerden biri haline gelmiştir. Veri kaynaklarının hızla artmasıyla farklı kaynaklardan ve farklı tiplerde veriler hayal edilemeyecek miktarda bir yer kaplamaya başlamıştır. Bu verinin depolanması, işlenmesi, analiz edilmesi ve yorumlanması gerekmektedir.

Büyük veri çözümleri eğitim, sağlık, ticaret, eğlence gibi birçok alanda kullanılmaktadır. İnternet sitelerinden ve sosyal medyadan elde edilen verilerin yorumlanmasında, bireylerin alışkanlıklarının ve kişiliklerinin çıkartılıp gruplama yapılarak öneri sistemlerinin oluşturulmasında, hasta bireylerden alınan bilgilerin anlamlandırılmasında, öğrencilerin öğrenme süreçlerinin tasarlanmasında ve uygulanan yöntemin değerlendirilmesinde büyük veri çözümlemelerinden faydalanılmaktadır.

Büyük veri, geleneksel veri işleme uygulamaları ile depolanamayacak, işlenemeyecek, analiz edilemeyecek kadar büyük ve karmaşık olan veri setleridir (Landset vd. 2015). Büyük veri, hacim, hız ve çeşitlilik olmak üzere üç ana bileşen ile ifade edilmektedir. Burada hacim veri büyüklüğünü temsil etmekte olup, gün geçtikçe bu veri büyüklüğü artmaktadır ve bir süre sonra veri büyüklüğü hayal edilemeyecek bir hal alacaktır. Hız, büyük verinin üretilme hızını ifade etmekte olup, verinin aynı hızda ulaşılabiliyor ve analiz edilebiliyor olması gerekmektedir. Çeşitlilik bileşeni ise verinin tipi olup, verinin çeşitli kaynaklardan çeşitli formatlarda gelmesini ifade etmektedir (Landset vd. 2015). Yapılan çalışmalar sonucunda büyük veri bileşenlerine değer ve doğrulama bileşenleri de eklenmiştir. Değer bileşeni, büyük veri analizinin pozitif yönlü bir değer katması gerekliliğini ifade ederken, doğrulama bileşeni verinin güvenli olması yani gizli kalarak doğru kişiler tarafından izlenmesi, görülmesi anlamına gelmektedir.

Büyük veri, birçok sorunu da beraberinde getirmektedir. Bu sorunlar, aşağıdaki gibi sıralanabilmektedir:

Ağ bant genişliği kapasitesi: Veriler genellikle dağıtık sistemlerde ve bulutta depolandığı için ağ bant genişliği yeterli gelmemektedir (Wang vd. 2016).

Veri güvenliği: Verinin güvenliği sağlanabilmeli, veri belirli kişilerle paylaşılmalıdır. Örneğin; sağlık kayıtları ve banka kayıtları önemli ve hassas bilgileri içermekte olup, bu verilerin gizliliği korunabilmelidir (Wang vd. 2016).

Verinin görselleştirilmesi: Veri çok boyutlu ve çok miktarda olduğundan, verinin görselleştirilebilmesi oldukça zordur (Wang vd. 2016).

Veri karmaşıklığı: Belirsiz, eksik ve dinamik veriler verinin karmaşık hale gelmesine neden olmaktadır (Wang vd. 2016).

Hesaplama karmaşıklığı: Verinin hacmi çok büyük olduğundan, kullanılan geleneksel metot ve araçlarla veri işlenememektedir (Wang vd. 2016).

Aykırı değerler: Verinin miktarı fazla olduğundan aykırı değerlerin tespiti zorlaşmaktadır (Anonymous 2017a).

Sonuçların gösterimi: Veri miktarı fazla olduğundan, sonuçların görselleştirilmesi efektif bir şekilde yapılamamakta ve doğru kararlar verme olasılığı azalmaktadır (Anonymous 2017a).

Senkronizasyon: Veriler birçok kaynaktan gelmektedir. Senkronizasyon sağlanmadığında tutarsız ve doğru olmayan bilgiler kullanılacak, bu da yanlış kararlar almaya neden olacaktır.

Veriye erişim: Artan hacim ve karmaşıklık veriye erişimi etkilemektedir.

Hadoop kütüphanesi ve Spark birleşik analiz motoru aracılığı ile büyük veri sorunlarının önüne geçilmesine çalışılmaktadır. Bu tez çalışmasında ağ bant genişliği kapasitesi,

hesaplama karmaşıklığı ve sonuçların gösterimi gibi sorunlarla karşılaşılmıştır. Bu sorunları çözmek adına büyük miktarda veri ve hesaplama içeren sorunları çözmeyi sağlayan Hadoop açık kaynak kodlu kütüphanesi ve makine öğrenimi algoritmalarının kullanımını sağlayan Mahout projesinden yararlanılmıştır.

Bu tez çalışmasının ilk bölümünde Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları kullanılarak büyük veri analizinde bulunulmuştur. Büyük veri çözümleri yardımıyla öneri sistemleri oluşturabilmekte ve öneri sistemleri performansı iyileştirilebilmektedir. Öneri sistemleri bir kullanıcının bir öğeye olabilecek olan ilgisini öngörmeye amaçlayan filtreleme sistemleridir ve kullanıcıların ürünler hakkındaki oylamaları ya da ürünlerin özellikleri gibi arkaplan verilerini kullanarak ürün ya da veri önermeyi amaçlamaktadır. Öneri sistemleri haber ve makale siteleri, e-ticaret siteleri gibi birçok alanda kullanılmaktadır. Bu tez çalışmasında mahout projesinden yararlanarak öneri sistemi performansı iyileştirilmeye çalışılmıştır.

Makine öğrenmesi, var olan veri üzerinden öğrenen ve tahmin sistemlerinde ve bilginin keşfinde kullanılan bir teknolojidir. Gözetimli ve gözetimsiz olmak üzere iki tip makine öğrenmesi algoritması bulunmaktadır. Gözetimli öğrenme sisteminde verilere ait etiket bilgisi yer almaktadır. Veri örneklerinin ise kendisine en benzer etikete atanması gerekmektedir. Gözetimli öğrenmeye sınıflama algoritmaları örnek olarak verilebilmektedir. Gözetimsiz öğrenmede ise etiket bilgisi yer almamaktadır. Benzer örnekler gruplanmakta ve ardından bu gruplara etiket bilgisi atanmaktadır. Gözetimsiz öğrenmeye örnek olarak kümeleme algoritmaları verilebilmektedir.

Apache Hadoop sistemi, büyük verinin hacim, hız ve çeşitlilik gibi zorluklarının üstesinden gelebilmektedir. Bu sistem MapReduce ve Hadoop Dağıtılmış Dosya Sistemi (HDFS) olmak üzere iki ana elementten oluşmaktadır. MapReduce elementi verinin paralel olarak işlenmesinde görevlidir. HDFS, yüksek bant genişliğine sahip hesaplama ve dağıtılmış düşük maliyetli depolama sağlamaktadır ve NameNode ve DataNodes olmak üzere iki bölümden oluşmaktadır. NameNode kısmı dosya sisteminin yönetiminden sorumlu iken, DataNode kısmı verinin bloklar halinde tutumundan sorumludur (Zheng, 2014).

Hadoop ekosistemi içerisinde Mahout, Zookeeper, Oozie, Flume, Sqoop, Chukwa ve Avro gibi bileşenler de bulunmaktadır. Bu bileşenlerden Mahout bileşeni sınıflama, kümeleme, işbirlikçi filtreleme gibi makine öğrenmesi uygulamalarından sorumludur. Mahout kütüphanesi lojistik regresyon, Naif Bayes ve Tamamlayıcı Naif Bayes gibi sınıflandırma algoritmalarına olanak sağlarken; gölgelik kümeleme, k-ortalama kümeleme, bulanık k-ortalama, akış k-ortalamaları, spektral kümeleme gibi kümeleme algoritmalarına olanak sağlamaktadır. Kullanıcı tabanlı ve ürün tabanlı işbirlikçi filtreleme de mahout kütüphanesinin sunduğu filtreleme yöntemlerinden biridir (Anonymous 2020a).

Makine öğrenmesi algoritmaları metinlerin ve farklı tipteki veri türlerinin sınıflandırılmasında kullanılan tekniklerden biridir. Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları kullanılan makine öğrenmesi tekniklerindendir. Kolay ve hızlı uygulamasından dolayı Naif Bayes algoritması en çok tercih edilenlerdendir. Bir diğer taraftan Naif Bayes algoritmasının zayıf tahmin yeteneği, karar sınırı için zayıf ağırlıkların seçilmesi, özelliklerin bağımsız olduğunu varsayması ve metinleri iyi modelleyememesi gibi problemleri de bulunmaktadır (Rennie, 2003).

Çalışmanın 3.2.1 kısmında Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizinde bulunulmuş, bu algoritmalarının ortalama doğruluk, ortalama kappa, ortalama güvenilirlik, ortalama ağırlıklı hassasiyet, ortalama ağırlıklı hatırlama, ortalama ağırlıklı F1 derecesi, ortalama eğitim ve test süreleri karşılaştırılmaları yapılmıştır.

Bu tez çalışmasında yer alan konulardan biri de duygu analizidir. Duygu analizi, bilimsel ve market alanlarında kullanılan en popüler konulardan biridir. Duygular tutum ya da görüş olabilmektedir. Duygu analizi, herhangi bir ürün, kişi, olay ya da kişi hakkındaki metinlere odaklanmakta ve bunun analizini yapmaktadır. Genelde, metinler pozitif-negatif, iyi-kötü, beğen-beğenme, düşük riskli-yüksek riskli gibi iki gruba sınıflandırılmaktadır. Bu sınıflara yeni gruplar da eklenerek duyguların derecesi çeşitlendirilebilmektedir. Örneğin; düşük riskli-yüksek riskli grubunun duygu derecesi artırılabilmekte ve orta riskli ve çok yüksek riskli gibi sınıflar eklenebilmektedir.

Duygu analizi fikir madenciliği olarak da adlandırılabilmektedir ve makine öğrenmesi ve doğal dil işlemenin ele aldığı konulardan biridir. İnsanlar bir ürün, etkinlik ya da fikir hakkındaki duygularını forumlar, mikro bloglar veya çevrimiçi sosyal ağ siteleri gibi sosyal medyayı kullanarak duygularını ifade etmektedir. Örneğin; bir ürün hakkında, siyasi ve dini görüşleri hakkında ya da kendi hayatları hakkında yazabilmektedirler.

Duygu analizi ile bazı problemler çözülebilmekte, birçok ihtiyaç karşılanabilmektedir. Örneğin, bir ürünle ilgili yorumların pozitif ya da negatif olduğunun belirlenmesinde, tweet verileri kullanılarak bir ürünün yeni sürümleri hakkındaki kullanıcıların tepkilerinin tespitinde, öznel verinin ayırt edilmesinde, düşünce tabanlı soruların tespitinde duygu analizi kullanılmaktadır.

3 farklı duygu analizi seviyesi bulunmaktadır: belge seviyesi, cümle seviyesi ve görünüş seviyesi. Belge seviyesi analizinde tüm doküman dikkate alınmakta ve o dokümanın hangi duyguyu anlattığı tespit edilmektedir. Cümle seviyesi analizinde her bir cümlenin ayrı ayrı analizi yapılmakta ve içerdiği duygu tespit edilmektedir. Görünüş seviyesi analizinde ise varlıkların özelliklerinin analizi yapılmaktadır (Medhat vd. 2014).

Duygu analizi makine öğrenmesi, kural tabanlı ve sözlük tabanlı yaklaşımlar olmak üzere 3 yaklaşımla yapılabilmektedir. Makine öğrenmesi yaklaşımında çıktıları bilinen girdiler aracılığı ile eğitim sağlanmakta ve model oluşturulmaktadır. Oluşturulan modelle yeni girdilerin çıktılarının tahmini yapılmaktadır. Destek Vektör Makinesi, N-gram Duyarlılık, Naif Bayes Metodu, Maksimum Entropi Sınıflandırıcısı, K-NN ve Ağırlıklı K-NN duygu analizinde kullanılan yöntemlerden biridir (Devika vd. 2016). Kural tabanlı yaklaşımda metinler seçilen kurala göre pozitif ve negatif kelimelerin sayısına göre analiz edilmektedir. Bu kurallar ince taneli sözlük, olumsuzluk kelimeleri, güçlendirici sözler olabilmektedir (Anonymous 2020b). Sözlük tabanlı olan analizde kelimelerin polarite değerleri toplamı dokümanın polarite değerini vermektedir (Kaushik ve Mishra 2014). Tez çalışmasının 3.2.2 kısmında Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile duygu analizinde bulunulmuştur.

Bu tez çalışmasında yapılan çalışmalardan biri de öneri sistemi geliştirilmesidir. Öneri sistemleri aşırı bilgi yüklemesi sorununun çözümü için geliştirilen, kullanıcıların ürünler hakkındaki oylamaları ya da ürünlerin özellikleri gibi arkaplan verilerini kullanarak ürün ya da veri öneren sistemlerdir. Aşırı bilgi yüklemesi, sistemin veri işleme kapasitesini aşacak kadar çok miktarda veri olduğunda meydana gelmektedir (Paradarami vd. 2017). Sonuç olarak, etkili bir karar almak güçleşmektedir (Speier vd. 1999). Öneri sistemleri öneriler sunarak karar vermeyi kolaylaştırmakta ve aşırı bilgi yüklemesi sorununu çözmektedir (Paradarami vd. 2017, Resnick ve Varian 1997, Ricci vd. 2011).

Öneri sistemleri öneride kullanılan teknikler ve bilgiler göz önüne alındığında dört gruba ayrılmaktadır. Bu gruplar işbirlikçi filtreleme, içerik tabanlı filtreleme, bilgiye dayalı tavsiye sistemleri ve hibrit öneri sistemleri olarak sıralanabilmektedir (Paradarami vd. 2017, Adomavicius ve Tuzhilin 2005, Burke 2002, Jannach vd. 2010).

İşbirlikçi filtreleme sistemleri yalnızca kullanıcıların oylarını kullanarak önerilerde bulunmakta (Yera ve Martinez 2017), birbirine benzer ilgi ve fikirleri olan kullanıcıların birbirine benzer ürünü tercih edeceğini varsaymaktadır (Paradarami vd. 2017). İçerik tabanlı filtreleme sistemleri ise kullanıcı tarafından geçmişte tercih edilen ve benzer özelliklere sahip olan ürünleri önermektedir (Paradarami vd. 2017). Bilgiye dayalı öneri sistemleri, önerilen ürünün özelliklerinin kullanıcının ihtiyaçlarını ne derecede karşıladığını göz önüne alarak önerilerde bulunurken (Paradarami vd. 2017, Ricci vd. 2011), hibrit öneri sistemleri iki ya da daha çok tekniği bir arada kullanarak önerilerde bulunmaktadır (Paradarami vd. 2017).

Gunawardana ve Shani (2009) tarafından yapılan çalışmaya göre öneri sistemlerinin tahmin ve öneri olmak üzere iki görevi vardır. Tahmin kısmında oylar tahmin edilmekte, öneri kısmında ise ilgili ürünler kullanıcıya önerilmektedir.

İşbirlikçi filtreleme öneri sistemleri soğuk başlangıç problemi, veri seyrekliği, ölçeklenebilirlik, eş anlamlı, gri koyun ve şilin saldırıları gibi bazı problemlerle karşı karşıyadır (Su ve Khoshgoftaar 2009). Soğuk başlangıç problemi, yeni bir kullanıcı için, o kullanıcı ile ilgili yeterli bilgiye sahip olunmadığından öneri yapılamaması durumunda

olmaktadır (Panigrahi vd. 2016, Gediminas ve Tuzhilin 2005). Ölçeklenebilirlik, gerçek zamanlı ya da gerçek zamana yakın bir zaman diliminde öneri yapabilme yeteneğidir (Panigrahi vd. 2016, Zhou vd. 2008, Isard ve Yu 2009). Veri seyrekliği problemi, sisteme yeni bir kullanıcı ya da ürün eklendiğinde, veri yetersizliği yüzünden iyi bir öneri yapılamamasıdır (Su ve Khoshgoftaar 2009). Eş anlamlılık problemi, öneri sistemlerinin birbirinin aynısı ya da çok benzeri olup farklı isimle temsil edilen ürünleri tespit edememesidir (Su ve Khoshgoftaar 2009). Gri koyun problemi, diğer insanlardan çok farklı birer zevke sahip olan kişiler yüzünden olmaktadır. Bu kişiler, öneri sisteminin performansını negatif olarak etkilemekte, kendileri için de iyi bir öneri yapılamamaktadır (Ghazanfar ve Prugel-Bennett 2011). Öneri sistemlerine yapılan saldırılar şilin saldırıları olarak adlandırılmaktadır ve bunlardan en genel olanları itme ve bomba saldırılarıdır. İtme saldırılarında, saldırgan sisteme ön yargılı puanlar ekleyerek istediği ürünün daha fazla önerilmesini amaçlamaktadır. Bomba saldırılarında ise istenilen ürünün daha az önerilmesi amaçlanmaktadır (Zhou vd. 2018).

Öneri sistemleri özellikle en çok e-ticaret sitelerinde ve haber sitelerinde kullanılmaktadır. Geliştirilen öneri sistemleriyle kişilere çeşitli öneriler sunulmaktadır. Örneğin; e-ticaret sitelerinde, kullanıcının diğer kullanıcılarla arasındaki ilişki hesaplanmakta, oluşan sonuca göre ürün önerisi yapılmaktadır. Ya da, ürünlerin diğer ürünlerle olan ilişkisi hesaplanmakta, oluşan sonuca göre kişiye yeni ürün önerilmektedir. Öneri sistemleriyle büyük veri yorumlanmakta, çeşitli çıkarımlar yapılmaktadır. Bu tez çalışmasının 3.2.3 kısmında öneri sistemi geliştirilmiştir.

Tez çalışmasında en son yapılan çalışma öneri sistemlerinin tahmin performansını iyileştirmek için yapay zeka tekniklerinden yararlanılarak evrimsel sinir ağlarının geliştirilmesi olmuştur. Yapay zeka, insan zekasını ve davranışını anlama ve taklit etme üzerinde kurulmuş bir alandır ve bilgisayarları insan gibi düşündürme yoluna gitmektedir. Yapay zeka teknikleri ile yargılama, muhakeme, kanıtlama, tanımlama, algılama, anlama, iletişim, tasarım, düşünme, öğrenme ve problem çözme gibi birçok aktiviteler gerçekleştirilebilmektedir. Yapay zeka teknikleri, uzman sistemler, makine öğrenmesi, örüntü tanıma, doğal dil anlayışı, otomatik teorem kanıtlama, otomatik programlama,

robotik, oyun teorisi, akıllı karar destek sistemleri ve yapay sinir ağları gibi birçok alanda kullanılabilmektedir (Xian 2010).

Ağın hızlı gelişimi bilginin aşırı büyümesine neden olmakta ve böylece yararlı bilgiye erişim zorlaşmaktadır. Bu durum, aşırı bilgi yükü problemi olarak adlandırılmaktadır. Bu problemi çözmek için arama motorları ve akıllı öneri sistemleri geliştirilmiştir. Arama motorları ve akıllı öneri sistemlerinin performansı karşılaştırıldığında, akıllı öneri sistemlerinin performansının daha iyi olduğu görülmektedir. Bunun nedenine örnek olarak şu durum verilebilmektedir: Kullanıcılar arama motoruna aynı anahtar kelimeyi girdiklerinde benzer sonuçlara ulaşacaklardır (Xiang vd. 2013, Guo-xia ve He-ping 2012). Sonuç olarak, bilgi erişim sistemleri kişisel ihtiyaçları karşılayamamaktadır. Akıllı öneri sistemleri ise kişinin ilgi alanına odaklanmakta, kullanıcı verilerini girdi olarak kullanmakta, kullanıcının ilgi alanını baz alarak önerilerde bulunmaktadır (Xiang vd. 2013).

Öneri sistemleri bir tür bilgi filtreleme sistemleri olup, kullanıcı tercihlerini, çevresel bağlamları ve kullanıcı bağlamlarını kullanarak mantıklı önerilerde bulunmayı amaçlamaktadır (Hirakawa vd. 2015). Öneri sistemleri günümüzde film, müzik, haber, kitap, sosyal etiket, ürün, restoran, finansal hizmetler, hayat sigortası, Facebook arkadaşları ve Twitter takipçileri önermek gibi durumlar için kullanılmaktadır (Jain vd. 2015) Örneğin; çevrimiçi video topluluğu olan YouTube sitesi, kullanıcının eski etkinliklerine dayalı olarak videolar önermektedir. İş odaklı sosyal ağ sitesi olan LinkedIn, kişiyi ilgilendirebilecek kişi, grup, topluluklar önermektedir. E-ticaret sitesi olan Amazon ise kitap, elektronik cihazlar gibi çeşitli ürünleri satmakta ve öneri sisteminde işbirlikçi filtreleme kullanmaktadır (Jain 2015, Jones 2013).

Öneri sistemleri içerik tabanlı filtreleme sistemleri, işbirlikçi filtreleme sistemleri, bağlama duyarlı öneriler, bilgiye dayalı öneriler, demografik filtreleme sistemleri ve hibrit önerici sistemleri olarak gruplanabilmektedir (Peerzade 2017). Bu öneri sistemlerinden içerik tabanlı öneri sistemi aktif kullanıcı ve ürünler hakkındaki bilgiyi kullanmakta (Jain vd. 2015) ve geçmişte tercih edilen ürünlere benzer ürünler önermektedir (Peerzade 2017). İşbirlikçi filtrele sisteminde ise aktif kullanıcıya öneride bulunulmasında, kullanıcılar ve

bu kullanıcılarla ürünler arasındaki ilişki bilgisi kullanılmaktadır (Jain 2015). İşbirlikçi filtreleme sistemleri kullanıcı tabanlı ve ürün tabanlı öneri sistemleri olmak üzere iki gruba ayrılmaktadır. Kullanıcı tabanlı öneri sistemlerinde hedef kullanıcılara en benzer ilgiye sahip olan kullanıcılar tespit edilirken, ürün tabanlı öneri sistemlerinde kullanıcıların davranışları analiz edilerek ürünler arasındaki ilişki tespit edilmektedir (Liu ve Wu 2019). Bağlam kelimesi bir kişinin, yerin ya da objenin durumunu analiz eden bilgi olarak tanımlanmaktadır (Mukherjee vd. 2011, Dey 2001). Bağlama duyarlı öneri sistemleri tahminlerinde kullanıcı oylarını ya da ürünlerin etkileşimlerini kullanmaktadır (Inzunza ve Juarez-Ramirez 2016, Codina vd. 2015). Bilgiye dayalı öneri sistemleri kullanıcılar ve ürünler hakkındaki bilgileri kullanmakta ve hangi ürünlerin kullanıcıların gereksinimlerini karşıladığını analiz etmektedir (Peerzade 2017). Demografik filtreleme sistemleri öneride bulunmak için kadınlar, erkekler, gençler gibi demogradik sınıfları kullanırken, hibrit öneri sistemleri iki veya daha fazla öneri tekniğini birleştirerek öneride bulunmaktadır (Peerzade 2017). Bu tez çalışmasının 3.2.4 kısmında öneri sistemlerinin tahmin performansını iyileştirmek için evrimsel sinir ağları kullanılmıştır.

1.1 Tezin Amacı

Tezin amacı aşağıda listelenmiştir:

- Değişen yumuşatma, normalizasyon, ağırlık parametreleri ve veri seti büyüklüğüne göre Naif Bayes ve Tamamlayıcı Naif Bayes sınıflama algoritmalarının performans değerlendirme kriterlerine göre performansını gözlemleyebilmek ve bu kriterlere göre en iyi performansı gösteren parametreleri elde etmek.
- "Doğruluk" performans değerlendirme kriterine ve veri seti büyüklüğüne göre en iyi performansı gösteren algoritmayı ve parametre değerlerini kullanarak duygu analizinde bulunmak.
- İşbirlikçi öneri sistemini kullanarak kullanıcı tabanlı bir öneri sistemi geliştirmek ve bu sistemin farklı benzerlik ölçütü karşısındaki performansını gözlemleyebilmek.
- Yapay sinir ağları ile film oy tahmininde bulunmak ve tahmin performansını genetik algoritma ile iyileştirebilmek.

olarak sıralanabilmektedir.

1.2 Tezin Özgün Değeri

Tezin özgün değeri;

- Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizi,
- Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının performansının detaylı analizi,
- Kullanıcı tabanlı olarak geliştirilen öneri sistemi ile bu sistemin performansının kullanılan farklı benzerlik hesaplama ölçütleri ile incelenmesi,
- Tahmin performansını artırmak adına melez bir sistemin geliştirilmesi ve genetik algoritmanın yapay sinir ağları tahmin performansını pozitif yönde etkilediğinin kanıtının yapılması.

olarak sıralanabilmektedir.

1.3 Tez Taslağı

Tezin 2. bölümünde literatür çalışmalarına yer verilmiştir.

Tezin 3. bölümünde tez çalışmasında yapılan deneylere ait olan materyal ve yöntemler hakkında bilgi verilirken, 4. bölümünde araştırma bulguları hakkında bilgi verilmiştir. Tezin 5. bölümünde ise genel sonuçlar yer almaktadır.

Yapılan çalışmalar arasında ilk olarak Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizi yer almaktadır. Naif Bayes ve Tamamlayıcı Naif Bayes sınıflama algoritmalarının değişen veri seti büyülüğü, normalizasyon, ağırlık ve yumuşatma parametreleri karşısında değişen ortalama doğru sınıflanan örnek yüzdesi, ortalama kappa, ortalama doğruluk, ortalama ağırlıklı hassasiyet, ortalama ağırlıklı hatırlama, ortalama ağırlıklı F1 derecesi, eğitim ve test süreleri gibi performans

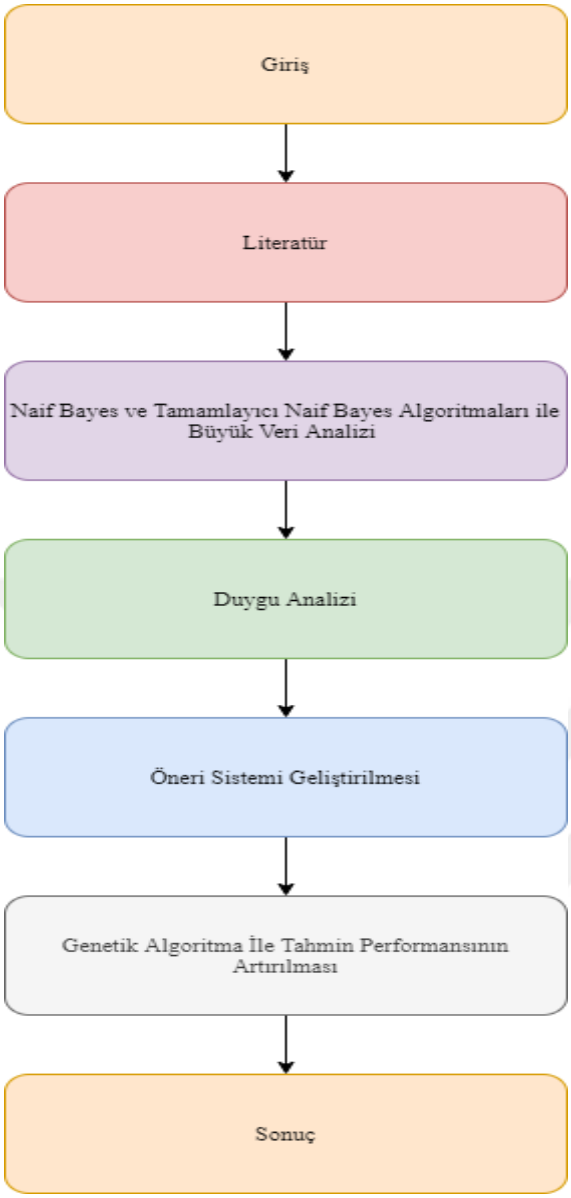
değerlendirme kriterleri gözlenmiştir. Tüm performans değerlendirme kriterleri için en iyi değeri gösteren ağırlık, yumuşatma, normalizasyon parametreleri ve veri büyüklüğü tespit edilmiştir.

Tez çalışmaları arasında ikinci olarak, ilk çalışmada elde edilen sonuçlardan yararlanılarak (her iki algoritma için veri setleri büyüklüğüne ve "doğruluk" performans değerlendirme kriterine göre tespit edilen en iyi parametre değerleri) Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile duygu analizinde bulunulmuştur. Filmler ve bu filmler hakkında yorumlar ve bu filmlere verilen oylar gibi kısımlardan oluşan bir veri setinin film yorumları kısmının pozitif, negatif ve nötr gibi sınıflaması yapılmıştır. Sonuç olarak, veri setlerine bir özellik daha eklenmiş olmuştur. Ardından, en doğru analiz değerine sahip olan veri seti kullanılarak tahmin edilen duygu ile o filme verilen oy oranı arasındaki ilişki Lojistik Regresyon algoritması ile incelenmiştir.

Yapılan tez çalışmalarından biri de kullanıcı tabanlı bir öneri sistemi geliştirilmesi ve geliştirilen öneri sisteminin değişen benzerlik hesaplama yöntemine göre değişen performansının gözlenmesi olmuştur. Benzerlik hesaplama yöntemi olarak Pearson, Öklid, log benzeri, merkezlenmemiş kosinüs gibi yöntemler kullanılmıştır.

Yapılan çalışmalar arasında son olarak yapay sinir ağları kullanılarak filmlere verilen oyların tahminin yapılması, ardından, sistem performansının genetik algoritma ile iyileştirilmesi yer almaktadır.

Tez kapsamını özetleyen bir akış şeması **şekil 1.1**'de verilmiştir.



Şekil 1.1 Tez kapsamı

2. LİTERATÜR TARAMASI

Bu çalışmada büyük veri öneri sistemleri üzerine çalışmalar yapılmıştır. Algoritma olarak yapay sinir ağları algoritmalarından yararlanılmıştır. Yapılan çalışmalarla ilgili olan literatür çalışmaları gruplanarak verilmiştir.

2.1 Büyük Veri Analizi

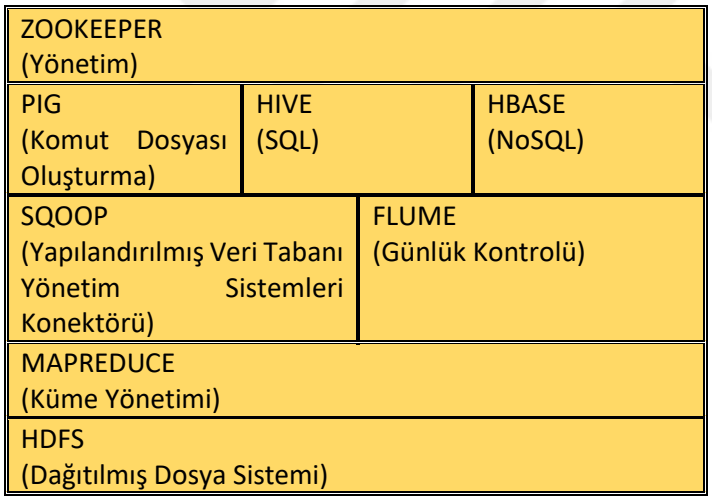
Büyük veri analizi sağlık, elektrik üretimi, radyosyon onkolojisi, multimedya verisinin analizini gibi birçok işlemde kullanılabilmektedir. Bu bölümde büyük veri analizinin yapıldığı çalışmalara yer verilmiştir.

Huang vd. (2015) tarafından yapılan bir çalışmada sağlık alanında büyük veri kullanım alanları, büyük veri depolanması, transferi ve analitik metotları özetlenmiştir.

Büyük verinin görselleştirilmesi, bu veriden yararlanılmasında önemli bir rol oynamaktadır. Iqbal vd. (2016) tarafından yapılan bir çalışmada medikal büyük verinin görselleştirilmesi için bir aracın geliştirilmesi amaçlanmıştır. Kanser hastalığının diğer hastalıklarla olan ilişkisini gözlemlemek için "Cancer Associations Map Animation (CAMA)", isimli bir araç geliştirilmiştir. Tayvan Ulusal Sağlık Sigortası Veritabanı'nda bulunan ve 782 milyon ayakta tedavi edilen hastaya ait olan bilgiler kullanılarak 9 başlıca kanser hastalıklarının kanser ve diğer hastalıklarla olan ilişkisi yaşa ve cinsiyete göre görselleştirilmiştir.

Büyük veri analizinin kullanıldığı tahmin sistemlerinden biri de elektrik üretimi tahmin sistemidir. Rahman vd. (2016) tarafından yapılan bir çalışmada elektrik üretimi tahmin sistemi geliştirilmiştir. Veri seti olarak Amerika'ya ait olup güç yönetimi üzerine toplanan ve son 20 yıla ait olan veriler kullanılmıştır. Verinin depolanması, ilgili verinin alınması ve yapılandırılmış formata çevrilmesi için Hadoop MapReduce kullanılmıştır. Ardından, geri beslemeli sinir ağları ile sistem eğitilmiş ve tahminde bulunulmuştur. Tahminlerde doğruluk oranı %99'u bulmuştur. Hadoop mimarisini gösteren bir figür **Şekil 2.1**'de

verilmiştir (Sachdeva vd. 2017). **Şekil 2.1**'de de görüldüğü üzere Hadoop mimarisi HDFS, MapReduce, SQOOP, FLUME, PIG, HIVE, HBASE, ZOOKEEPER gibi bölümlerden oluşmaktadır. Bu bölümlerden HDFS verilerin güvenilir ve yükseltilebilir bir şekilde depolanmasını sağlarken, MapReduce ile birçok iş paralel olarak yürütülebilmektedir. Sqoop aracılığıyla Hadoop ile diğer farklı türdeki veri tabanları arasında veri aktarımı gerçekleştirilebilmektedir. Flume ise çeşitli kaynaklardan gelen büyük miktardaki veriyi depolayıp kümelendirerek ve ardından bu büyük miktardaki veri günlüklerini atanan havuz alanlarına taşıyarak çalışan bir servistir. Pig bir veri akışı dili olup paralel hesaplama yöntemlerinde kullanılırken, hive verilerin ve işlenen işlerin sonuçlarının depolanması için bir depo ortamı sağlamaktadır. HBase No-SQL veri tabanı olup, kullanıcılara büyük verinin gerçek zamanlı kullanılabilirliğini sağlamaktadır. Zookeeper ise Hadoop kümesindeki bir sorunu izleme ve düzeltme sorumluluğunu üstlenmiştir (Sachdeva vd. 2017).



Şekil 2.1 Hadoop ekosistemi (Sachdeva vd. 2017)

Radyasyon onkolojisi konusunda da büyük veri analiz tekniklerinden yararlanılabilmektedir. Bibault vd. (2016) tarafından yapılan bir çalışmada "radiation therapy," "bioinformatics," "big data", "genomics", "electronic health records", "decision support systems" ve "machine learning" anahtar kelimeleri kullanılarak 1980 yılının ocak ayından, 2016 yılının nisan ayına kadar olan bir tarih aralığını kapsayacak şekilde bir literatür çalışması yapılmıştır. Bu literatür çalışmasıyla radyasyon onkolojisi konusunda

bir tıp programının uygulanmasında yaşanan bilişim sorunlarını açıklamak ve sorunların üstesinden gelmek için uygulanan yöntemleri tanımlamak amaçlanmıştır. Sonuç olarak, radyasyon onkolojisi çalışmalarında en çok destek vektör makinesi, yapay sinir ağları çeşitleri ve derin öğrenmenin kullanıldığı gözlemlenmiştir. Radyasyon onkolojisi konusunda büyük veri analiz tekniklerinin kullanılma amacı ise veri tiplerinin çok çeşitli olması ve verinin hızlı bir şekilde analiz edilip istenilen formata çevrilmesidir.

Hastalık teşhisi, elektronik sağlık sistemi önerimi ve salgın hastalık tahmini gibi işlemlerde de büyük veri analizi tekniklerinin önemli bir katkısı yer almaktadır. Örneğin; Pasupathi ve Kalavakonda (2016) tarafından yapılan bir çalışmada hastanın şu anki test sonuçları ve geçmiş zamanlardaki tıbbi geçmiş bilgileri baz alınarak büyük veri teknikleri ile hastalık teşhisi ve buna bağlı olarak ilaç önerisi yapılırken, Ilapakurti vd. (2016) tarafından yapılan bir çalışmada bulut tabanlı elektronik sağlık sistemi önerilmiştir. Bu çalışmada gerçek zamanlı büyük veri işlemesinin sağlanması için Apache Kafka ve Apache Spark kullanılmıştır. Salgın hastalık tahmini sistemini geliştiren çalışmalara örnek olarak Chen vd. (2017) tarafından yapılan çalışma verilebilmektedir. Chen vd. (2017) tarafından yapılan bir çalışmada hastalığa sık rastlanan toplumlarda salgın hastalık tahminini makine öğrenmesi algoritmaları ile gerçekleştirilmiştir. Veri seti olarak 2013-2015 yılları arasını kapsayan ve Çin'de bulunan bir hastaneden alınan 31919 adet hastaya ait olan 20320848 adet kayıt kullanılmış ve serebral enfarktüs hastalığı tahmini yapılmıştır. Yapılan tahminler sonucunda yapılandırılmış ve yapılandırılmamış veriler ile konvolusyonel sinir ağı ile yapılan tahmin oranı %94.8 ile diğer veri tipleri için kullanılan naif bayes, k-en yakın komşuluk ve karar ağacı algoritmalarından daha doğru tahminde bulunmuştur.

Büyük veri analizinde kullanılan algoritmaların başında makine öğrenmesi algoritmaları gelmektedir. Zhou vd. (2017) tarafından yapılan bir çalışmada büyük veri üzerine makine öğrenmesi algoritmaları uygulamanın artı ve eksileri verilmiştir. Makine öğrenmesi sayesinde büyük veri üzerinde öğrenme daha doğru bir oranda gerçekleşmekte ve daha doğru tahminler yapılabilmektedir. Yalnız, büyük veri olduğunda karmaşıklık seviyesi artmaktadır. Örneğin, yeni sınıf ve kümelerin keşfi için makine öğrenmesi modelinin kapasitesinin artırılması gerekmektedir.

Multimedya verisinin analizini optimum bir hızda ve optimum bir kaynak tahsisi olacak şekilde yapılabilmesi de büyük veri analizi ile gerçekleştirilebilmektedir. Jayasena vd. (2017) tarafından yapılan bir çalışmada multimedya verisinin analizini optimum bir hızda ve optimum bir kaynak tahsisi olacak şekilde yapılabilmesi amaçlanmıştır. Multimedya verisi farklı kaynaklardan ve farklı tiplerden gelen verinin aynı semantiği temsil ettiği veridir. Örneğin; bir web sayfasında görüntü ya da videolar bir metin ya da etiket ile ilişkilendirilerek verilmektedir. Bu gibi farklı türdeki veriler multimedya verisini oluşturmaktadır. Servis, platform ve alt yapı katmanı olmak üzere 3 katmandan oluşan bir mimari önerilmiştir. Hizmet katmanında video dosyaları toplanmış, dosyalar çeşitli dijital araçlar için uyumlu dosyalar haline getirilmiştir. Platform katmanında veriler Hadoop + MapReduce ile MPEG-4 formatına çevrilmiştir. Alt yapı katmanında ise optimum bir kaynak tahsisi için arı kolonisi algoritması kullanılmıştır.

2.2 Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları İle Büyük Veri Analizi

Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları ile büyük veri analizi yapan ve bu algoritmaların performansını karşılaştıran birçok çalışma bulunmaktadır. Bunlardan bazıları bu bölümde özetlenmiştir.

Kibriya vd. (2004), Jiang vd. (2013), Anagaw ve Chang (2018), Komiya vd. (2011) ve Keiser ve Dietterich (2009) Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının performansını karşılaştırmıştır. Örneğin; Kibriya vd. (2004) Çok Terimli Naif Bayes, Ağırlık-Normalize Tamamlayıcı Naif Bayes sınıflayıcı ve destek vektör makineleri algoritmalarının performansını karşılaştırırken ve performansı artırmak adına TFIDF skorunu kullanırken, Jiang vd. (2013) Çok Terimli Naif Bayes, Tamamlayıcı Naif Bayes, bire karşı hepsi bir arada model, Yerel Ağırlıklı Çok Terimli Bayes, Yerel Ağırlıklı Tamamlayıcı Naive Bayes ve Yerel Ağırlıklı Bire Karşı Hepsi Bir Arada algoritmalarının doğruluk performansını karşılaştırmıştır. Diğer yandan, Anagaw ve Chang (2018) Tamamlayıcı Naif Bayes ve Naif Bayes algoritmalarının performansını karşılaştırırken, Komiya vd. (2011) Naif Bayes Olumsuzluğu algoritması performansını Naif Bayes ve Tamamlayıcı Naif Bayes Algoritması performansıyla karşılaştırmıştır. Ayrıca, Keiser ve Dietterich (2009) Bernoulli Naif Bayes, Çok Terimli Naif Bayes, Dönüştürülmüş Ağırlık-

Normalize Tamamlayıcı Naif Bayes algoritmaları performaslarını karşılaştırmıştır. Bu çalışmalar sonunda benzer sonuçlara erişilmiştir. Örneğin; Kibriya vd. (2004) Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmalarının performanslarının veri setine göre değiştiğini ve performanslarının birbirine çok yakın olduğunu iddia etmişlerdir. Anagaw ve Chang (2018) Tamamlayıcı Naif Bayes Algoritmasının hem doğruluk hem de hesaplama zamanı performansının Naif Bayes Algoritması'nın performansından daha iyi olduğunu gözlemiştir. Komiya vd. (2011) veri eşit dağılmamış olduğunda Naif Bayes Olumsuzluğu algoritmasını önermiştir. Keiser ve Dietterich (2009) doğrusal güven ağırlıklı sınıflandırıcılar performansının daha iyi olduğunu gözlemlemiştir.

Naif Bayes Algoritması'nın performansını diğer algoritmalarla karşılaştıran çalışmalar da bulunmaktadır. Örneğin; Goyal ve Mehta (2012) Naif Bayes algoritmasının performansını J48 Algoritması ile karşılaştırırken, Ashari vd. (2013) karar ağacı ve k-en yakın komşu algoritmalarıyla karşılaştırmıştır. Friedman ve Goldszmidt (1996) Naif Bayes sınıflayıcı, ağaçla zenginleştirilmiş Naif Bayes, düzleştirilmiş ağaçla zenginleştirilmiş Naif Bayes, karar ağacı sınıflandırıcı (C4.5), seçici Naif Bayes sınıflandırıcı algoritmalarının doğruluk performanslarını karşılaştırmıştır. Nithya vd. (2015) ise Bayes Net, Naif Bayes, Naif Bayes çok terimli metin algoritmalarının performasını karşılaştırmıştır. Bunlara ek olarak Mussa vd. (2015), konik Naif Bayes algoritmasının performansını Laplacian düzeltilmiş değiştirilmiş Naif Bayes ve standart Naif Bayes algoritmalarıyla karşılaştırmıştır. Bu çalışmalardan sonra bazıları (Ashari vd. (2013), Friedman ve Goldszmidt (1996), Nithya vd. (2015)) Naif Bayes Algoritmasının daha iyi olduğunu gözlemlerken, bazıları J48 algoritmasını (Goyal ve Mehta (2012)) ve Konik Naif Bayes algoritmasını (Mussa vd. (2015)) önermiştir.

2.3 Duygu Analizi

Duygu analizi de öneri sistemleri sonucunda oluşan bir analizdir ve literatürde duygu analizi çalışmaları için kullanılan birçok makine öğrenimi algoritmaları bulunmaktadır. Xia vd. (2011), Zhang vd. (2011), Tan ve Zhang (2008), Ye vd. (2009), Smeureanu ve Bucur (2012) metin sınıflaması için Naif Bayes algoritmasını kullanmışlardır. Buna ek olarak Xia vd. (2011), Zhang vd. (2011), Tan ve Zhang (2008) ve Prabowo ve Thelwall

(2009) destek vektör makineleri algoritmalarını kullanmışlardır. Bu çalışmada ise duygu analizi için Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarından yararlanılmıştır.

Woldemariam (2016) tarafından yapılan çalışmada ise medya analizi çerçevesinde duygu analizi yapılmış, bu analiz için sözlük tabanlı yaklaşımlar ve makine öğrenimi algoritmaları kullanılmıştır. Sözlüğe dayalı duyarlılık tahmin algoritmasında hadoop çerçevesi, özyinelemeli sinir tensör ağı modeli için Stanford coreNLP kütüphanesi katkı sağlamıştur. Çalışma sonunda pozitif yorumların sınıflanmasında sözlük tabanlı yaklaşımların daha iyi performans gösterdiği görülmüştür.

Duygu analizi işlemlerinde makine öğrenimi algoritmaları ve sözlük tabanlı yaklaşımlar dışında yöntemler de bulunmaktadır. Bu yöntemlerin başında kelime vektörü, kısmı metin girişimi, duygu yayılımı ve çoklu duygu sözlükleri ve anlamsal kural kümeleri gibi yöntemler gelmektedir. Örneğin; Fan vd. (2016) tarafından yapılan çalışmada duygu analizi kelime vektörü kullanılarak yapılmıştır ve %85.77 F1 değeri, %85.20 hatırlama değeri ve %86.35 doğruluk değerine ulaşılmıştır. Gupta vd. (2019) tarafından yapılan çalışmada ise tweet'ler arasındaki anlamsal benzerliği bulmak ve onları gruplandırmak için metin girişimi yaklaşımının kullanılması önerilmiştir. Duygu yayılımı yaklaşımlarından yararlanarak Twitter mesajlarının analiz edilmesini sağalayan çalışma Wang vd. (2020) tarafından gerçekleştirilirken, çoklu duygu sözlükleri ve anlamsal kural kümeleri yöntemleri kullanılarak Çin mikro-blog duygu analizi gerçekleştiren bir algoritmayı öneren ve bu algoritmayla pozitif, nötr ve negatif olarak Çin mikro bloğunun doğru bir şekilde sınıflandırmasını gerçekleştiren çalışma Wu vd. (2019) tarafından yapılmıştır.

2.4 Öneri Sistemi Geliştirilmesi

Öneri sistemi geliştirilmesi ile ilgili olarak birçok çalışma yapılmıştır. Bu çalışmaların çoğunda genetik algoritma ve bulanık mantık algoritmalarının olduğu görülmektedir.

Siddiquee vd. (2014) bulanık mantık kullanarak bir öneri sistemi geliştirmişlerdir. Benzerlik ölçütü olarak Öklid Uzaklığı, Manhattan Uzaklığı, Pearson Katsayısı ve Kosinüs

Benzerliği ölçütlerini kullanmışlardır. Sonuç olarak Öklid Uzaklığı benzerlik ölçütünün en iyi sonucu verdiği gözlenmiştir. Parvin vd. (2018) tarafından yapılan diğer bir çalışmada ise genetik algoritma ve güven beyanı kullanılmıştır. Sonuç olarak, TCFGA olarak adlandırılan iki aşamalı Ortak Filtreleme metodunun daha iyi bir performans gösterdiği gözlenmiştir. Alhijawi ve Kilani (2016) genetik algoritma kullanarak "Simgen" adında bir sistem geliştirmişlerdir. Kullanıcılar arasındaki benzerlik genetik algoritma ile hesaplanmıştır. Tahmin kalitesi ve performansında sırasıyla %46'lık ve %38'lik bir gelişme sağlanmıştır. Bir diğer taraftan Verma vd. (2013) ve Jeon vd. (2009) tarafından yapılan çalışmada bulanık mantık kullanılmıştır. Verma ve arkadaşları işbirlikçi filtreleme ve bulanık c-ortalama kümelemesi algoritmalarını kullanarak bir öneri sistemi geliştirmişler, bulanık kümeleme algoritmasının k-ortalama metodundan daha iyi bir performans sergilediği gözlenmiştir. Jeon ve arkadaşları tarafından yapılan çalışmada işbirlikçi filtreleme ve bulanık sistem tabanlı bir sistem geliştirilmiştir. Sonuç olarak, bulanık sistemin işbirlikçi sistemin eksikliklerini tamamladığı görülmüştür.

Genetik algoritma ve bulanık mantık algoritmalarının bir arada kullanıldığı çalışmalar da bulunmaktadır. Örneğin; Shivhare vd. (2015) bulanık c-ortalama kümelemesi ve genetik algoritma tabanlı ağırlıklı benzerlik ölçüsü kullanarak bir öneri sistemi geliştirmişlerdir. Sonuç olarak, bulanık c-ortalama kümelemesi ve genetik algoritma ile ağırlaştırılmış benzerlik ölçütü kullanımının, genetik algoritma ile ağırlıklandırılmış benzerlik ölçütü kullanımından daha iyi bir sonuç verdiği gözlenmiştir.

Ölçeklenebilirlik, öneri sisteminin gerçek zamanda ya da gerçek zamana yakın bir zaman diliminde öneri yapabilme kabiliyeti olarak tanımlanmaktadır (Panigrahi vd. (2016), Zhou vd. (2008), Isard ve Yu (2009)). Shou-qiang ve Ming (2016) ölçeklenebilirlik problemi ile ilgili çalışmalar yapmış, Hadoop Mapreduce tabanlı paralel dönüşüm önermişlerdir. Ayrıca, hibrit işbirlikçi filtreleme kullanarak genetik algoritma tabanlı Hadoop küme öneri sistemi geliştirmişlerdir. Çalışmaların sonunda performansta iyileşmenin olduğu gözlenmiştir.

Genetik algoritma, içerik tabanlı filtreleme sistemlerinde de kullanılmaktadır. Örneğin; Kim vd. (2010) içerik tabanlı filtreleme sistemleri ve genetik algoritma kullanarak bir öneri sistemi geliştirmişlerdir. Sonuç olarak iyi bir performans gözlenmiştir.

Kullanıcılar ya da ürünler arasındaki benzerliği hesaplamak için birçok benzerlik hesaplama ölçütü bulunmaktadır. Bunların başında kosinüs bazlı benzerlik, korelasyon bazlı benzerlik, düzeltilmiş kosinüs benzerliği, Öklid uzaklığı, Manhattan uzaklığı, Minkowski uzaklığı, Pearson korelasyonu, log benzeri gelmektedir. Siddiquee vd. (2014) tarafından bildirildiğine göre Sarwar vd. (2001) kosinüs tabanlı benzerlik, korelasyon tabanlı benzerlik ve düzeltilmiş kosinüs benzerliği tekniklerini kullanarak ürün tabanlı bir öneri sistemi geliştirmişlerdir. Siddiquee vd. (2014) tarafından bildirildiğine göre Roy ve Kundu (2013) ise işbirlikçi filtreleme ve kümeleme kullanarak bir öneri sistemi geliştirmişler, kullanıcılar arasındaki benzerlikleri Öklid, Manhattan ve Minkowski uzaklıklarını kullanarak ölçmüşlerdir.

Kümeleme algoritmaları öneri sistemlerinde genellikle benzerliği ve en yakın komşuluğu bulmak için kullanılmaktadır. Verma vd. (2013) tarafından bldirildiğine göre Li ve Zhou (2003) ve Gong (2008) çalışmalarında kümeleme algoritmalarını kullanmışlardır. Li ve Zhou (2003) çalışmalarında içerik tabanlı ve işbirlikçi filtreleme tekniklerini kullanmışlardır. K ortalamalar kümeleme algoritması ile benzer kullanıcılar bulunmuş, ardından, aynı küme içinde bulunan ve kullanıcılar tarafından oylanan içerikler tesbit edilmiştir. Son olarak da, içeriği içerik listesine eklemişler ve aynı küme içerisinde talep edilen içeriği bulmak amacıyla bulanık c-means algoritması kullanmışlardır. Gong (2008), bulanık benzer öncelikli karşılaştırma ve bulanık kümeleme tabanlı bir öneri sistemi geliştirmiştir. Bulanık benzer öncelikli karşılaştırma kullanıcılar arasındaki benzerliği bulmak için kullanılırken, bulanık kümeleme algoritması en yakın komşuluğu bulmak için kullanılmıştır. Sonuç olarak, sistem performansında gelişmeler görülmüştür.

Düğüm sayısının hıza olan etkisini gözlemleyen çalışmalar da bulunmaktadır. Örneğin; Zhao ve Shang (2010) tarafından yapılan bir çalışmada 2.64 milyon film, 17000 kullanıcı içeren Netflix veriseti kullanılmıştır. Bu verisetinin 100, 200, 500 ve 1000'er kullanıcı içeren 5 kopyası alınmıştır. DataNode 2, 4, 6 ve 8 düğüme bölünmüştür. Tüm kullanıcıların

tüm filmlere vermiş olduğu oylar matris halinde temsil edilmiştir. Kosinüs benzerlik ölçüsü ile kullanıcılar arasındaki benzerlik hesaplanmıştır. Ardından, oy tahmini yapılmıştır ve film öneri listesi oluşturulmuştur. Algoritma 1 tanesi NameNode, 8 tanesi de DataNode olmak üzere herbiri 4GB Intel Dual-core 2.6 GHZ CPU olan 9 adet bilgisayarla yapılmıştır. Süre ölçümü yapılmıştır. Düğüm sayısı artınca hız da lineer olarak artmıştır. Veri miktarı küçük olduğunda bu durumun gözlenmeyebileceği yorumunda bulunulmuştur.

Veri seti büyüklüğünün hıza olan etkisini inceleyen çalışmalardan biri Jiang vd. (2011) tarafından yapılan çalışmadır. Bu çalışmada, 943 kullanıcı, 1670 film ve 54000 oylamadan oluşan MovieLens veri seti kullanılarak film öneri sistemi geliştirilmiştir. Tüm işlemler, veri büyüklüğüne bağlı olarak artan hesaplama süresini kısaltmak adına Hadoop üzerinde yapılmıştır. Bu çalışma 4 adet Map-Reduce bölümünden oluşmuştur. Map1-Reduce1'de her bir film için ortalama oy değeri hesaplanmıştır. Map2-Reduce2'de filmler arasındaki benzerlik hesaplanmıştır. Map3-Reduce3'te benzerlik matrisi, oylar ve her bir film için ortalama oy değerleri birleştirilmiştir. Map4-Reduce4'te oy tahmininde bulunulmuştur ve öneri listeleri oluşturulmuştur. Sistemin geliştirilmesi için öğe tabanlı işbirlikçi filtreleme algoritması kullanılmıştır. Benzerlik hesaplaması için de ise Pearson korelasyon ölçümü kullanılmıştır. Donanımsal olarak her biri Intel P4 CPU, 1G RAM, 80G disk olan 3 bilgisayar kullanılmıştır. Bilgisayarlar birbirine 100 Mbps anahtar ile bağlanmıştır. Performans değerlendirme kriteri olarak hızlandırma ve iso verimliliği kullanılmıştır. Veri seti büyüklüğü arttıkça hızlandırma değerinin de arttığı gözlenmiştir. İleriki çalışmalarda, map-reduce bölümündeki kaynak tahsisi ve yürütme sürecinin optimize edilmesi planlanmıştır.

Wang vd. (2014) tarafından ve Lin vd. (2014) tarafından yapılan çalışmada hibrit öneri sistemi geliştirilmiştir. Wang vd. (2014) tarafından yapılan çalışmada hibrit öneri algoritması içerik bazlı filtreleme algoritması ve işbirlikçi filtreleme algoritmalarının birleşiminden oluşmuştur. 1 tanesi ana düğüm, geriye kalan 7 tanesi veri düğümü olarak kullanılmak üzere toplam 8 adet bilgisayar kullanılmıştır. Tüm bilgisayarların CPU değeri Pentium Dual Core Processor 2.66 GHZ. olup her birinin harddisk kapasitesi değeri 80 GB'dır. Veri seti olarak MovieLens, performans değerlendirme kriteri olarak ortalama mutlak hata kullanılmıştır. Sonuç olarak veri seti büyüklüğü arttıkça performansın da

arttığı gözlenmiştir. Bilgisayar sayısının artması uygulama süresinin azalmasını sağlamıştır. Lin vd. (2014) tarafından yapılan çalışmada ise hibrit sistem işbirlikçi filtreleme, k-ortalama kümeleme ve eğim bir algoritmasından oluşmuştur. Eğim bir algoritması, kullanıcı tabanlı ve öğe tabanlı işbirlikçi algoritmalarına göre daha basit ve etkilidir. Fakat onun dezavantajı birçok kullanıcı tarafından oylanan ürünlerin oylanmasını hedeflemesidir. Kümeleme algoritması, bir kullanıcının diğer kullanıcılarla arasında bir benzerlik yoksa bu kullanıcının ürünleri nasıl oylayacağının tahminini kolaylaştırmak için kullanılmıştır. Movielens 100 K veri seti kullanılmış, deneyler 1 ana, 2 köle ve 1 ana 3 köle bilgisayarla yapılmıştır. Bilgisayarların hepsi de Intel Core i3 CPU 2.93 GHZ 4 GB hafıza 500 GB disk özelliklerine sahiptir. Hadoop versiyonu 1.0.3, mahout versiyonu 0.7'dir. K-ortalama kümeleme için k değerleri 5, 10, 15, 20, 25 olarak seçilmiştir. Seri hibrid tavsiye algoritması ve 2 ve 3 düğümlü paralel hibrid tavsiye algoritmasının performansını 5, 10, 15, 20, 25 küme ile karşılaştırılmıştır. Herbir algoritmada küme sayısı arttıkça zaman tüketimi artmıştır. Düğüm sayısı yani bilgisayar sayısı artınca zaman tüketimi azalmıştır.

Verma vd. (2015) tarafından yapılan bir çalışmada, Movielens veri setine ait olan farklı büyüklüklerdeki veri setleriyle mahout kütüphanesi aracılığı ile önerilerde bulunulmuştur. Veri setlerindeki büyüklük artsa da, uygulama süresinin aynı oranda artmadığı gözlenmiştir.

Subramaniyaswamy vd. (2015) tarafından ve Riyaz ve Varghese (2016) tarafından yapılan çalışmalarda işbirlikçi filtreleme yöntemi kullanılmıştır. Subramaniyaswamy vd. (2015) veri seti olarak twitter veriseti kullanmışlardır. Bu veri seti konum, bölüm, tweet, ifade gibi kısımlardan oluşmaktadır. Duygu analizi ile kullanıcı duyguları tahmin edilmiştir. İşbirlikçi filtreleme için girdi olarak tweet ve duygular verilmiştir. Kullanıcıların tahminlerine göre, tweet önerimi yapılmıştır. Hadoop framework ve Eclipse kullanılmıştır. Riyaz ve Varghese (2016) tarafından yapılan çalışmada ise Amazon veriseti kullanılarak işbirlikçi filtreleme yöntemi ile Hadoop MapReduce üzerinde roman önerimi gerçekleştirilmiştir. Gerekli olan veri amazon verisetinden alınmıştır. Veri; <Müşteri kimliği, ürün başlığı, oy> haline getirilmiştir. Pearson korelasyon katsayısı ölçümü ile kullanıcı tercihi bazlı benzer romanlar bulunmuştur. Öğe-öğe işbirliğine dayalı filtreleme ile roman önerimi gerçekleştirilmiştir. Donanımsal olarak Intel Pentium Dual Core 2 GHZ

işlemci ile 4 GB 1066 MHZ RAM kullanılmıştır. İşletim sistemi olarak ubuntu 15.10, hadoop versiyonu olarak 2.6.0 kullanılmıştır. Sonuç olarak veri büyüklüğü arttıkça Hadoop performansı ve veri düğümlerinin performansının arttığı gözlenmiştir.

2.5 Genetik Algoritma ve Yapay Sinir Ağları İle Tahmin Performansının Artırılması

Genetik algoritma kullanarak tahmin performansını artırma yoluna giden birçok çalışma bulunmaktadır.

Ar ve Bostancı (2016) benzerlik ağırlıklarını ayarlamak için genetik algoritmayı kullanmışlar ve doğruluk oranında iyileşme gözlemlemişlerdir. Kim vd. (2010) ve Inoue vd. (2016) müzik alanı için bir öneri sistemi önermişlerdir. Kim vd. (2010) yaptıkları çalışmada kullanıcıların tercihlerindeki değişikliklere hızlı bir şekilde adapte olmak ve cevaplamak için genetik algoritma ve içerik tabanlı filtreleme tekniğinin birleştirmişlerdir. Inoue vd. (2016) ise müzik öneri sistemi için dağıtık genetik algoritma önerisinde bulunmuşlardır. Soliman vd. (2015) ziyaret edilmemiş bölgeler için kullanıcıların ilgilerini tahmin etmek için genetik algoritma kullanmışlardır. Hassan ve Hamada (2017) ile Hamada vd. (2018) ise çok kriterli öneri sistemlerinin doğruluk tahmin performansını artırmak amacıyla genetik algoritma kullanmışlardır. Agrawal ve Jain (2017) öneri sistemlerinde vektör destek makineleri ve genetik algoritmayı kullanmışlar ve doğruluk, ölçeklenebilirlik ve kalite alanarında iyileşme gözlemlemişlerdir. Hosseinpourpia ve Oskoei (2017) çok yönlü güven modelinin parametrelerinin tahmininde genetik algoritmayı kullanmışlardır. Janjarassuk ve Puengrusme (2019) ürün öneri sistemini geliştirmek için genetik algoritmayı kullanırken, Ciaramella vd. (2010) kullanıcıya daha doğru kaynakları önerebilmek için genetik algoritma kullanmışlardır.

Öneri sistemlerinin performansının artırılması konusunda yapay sinir ağları da birçok çalışmada kullanılmıştır. Örneğin; Devi vd. (2010) kullanıcılar arasındaki güvene odaklanmış ve çalışmalarında olasılık sinir ağını kullanmışlardır. Hassan ve Hamada (2017) oy kriterlerini modellemek ve yapay sinir ağlarının çok kriterli öneri sistemlerine performansını analiz etmek için yapay sinir ağlarını kullanmışlardır. Chakrabarti ve Das

(2019) çalışmalarında işbirlikçi filtreleme sistemleri için kullanılan sinir ağları metotlarını incelemişlerdir. Da'u ve Salim (2019) ürünlerin ve kullanıcı özelliklerinin duygularına dayalı bir sinirsel dikkat ile duyarlı bir derin tavsiye sistemi önermişlerdir. Gong ve Ye (2009) geri beslemeli yapay sinir ağı ve ürün tabanlı işbirlikçi filtreleme tabanlı bir öneri sistemi önermişlerdir. Bu çalışmada boş oylar geri beslemeli sinir ağları ile doldurulurken, en yakın komşuluk ürün tabanlı işbirlikçi filtreleme ile hesaplanmıştır. Mai vd. (2009) çalışmasında sinir ağları kümelemeli tabanlı işbirllikçi filtreleme algoritma sistemi üzerinde durmuşlardır. Almaghrabi ve Chetty (2018) kullanıcılar ve ürünler arasındaki gizli ve saklı etkileşimleri tespit etmek amacıyla derin öğrenme tabanlı çerçeve örnek öneriminde bulunmuşlardır. Sanandaj ve Alizadeh (2018) ise işbirlikçi filtreleme teknikleri, içerik tabanlı filtreleme ve yapay sinir ağları tabanlı bir öneri sistemini önermişlerdir.

3. MATERYAL VE YÖNTEM

Bu bölümde tez çalışmasında yapılan Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizi, duygu analizi, öneri sistemi geliştirilmesi ve öneri sistemlerinin tahmin performansını iyileştirmek için evrimsel sinir ağları geliştirilmesi gibi deneylere ait olan materyal ve yöntemler hakkında bilgi verilmiştir.

3.1 Materyal

Bu bölümde tez çalışmasında yapılan deneylere ait olan materyal bilgisine yer verilmiştir.

3.1.1 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizi

Çalışmanın bu kısmında bu deney için kullanılan veri setleri, algoritmalar, kütüphaneler, performans değerlendirme kriterleri hakkında bilgiler verilmiştir. Veri seti olarak 20 Newsgroups (Anonymous 2008) veri seti tercih edilirken, algoritma olarak Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları tercih edilmiştir. Analiz işlemi Hadoop üzerinde mahout kütüphanesi ile yapılmış; işlemci teknolojisi olarak Core i7, işlemci çekirdek sayısı 4, işlemci hızı 2.0 GHz olan bir bilgisayar seçilmiş, performans değerlendirme kriteri olarak ortalama doğruluk, ortalama kappa, ortalama ağırlıklı hassasiyet, ortalama ağırlıklı hatırlama, ortalama ağırlıklı F1 derecesi, ortalama eğitim ve test süreleri kullanılmıştır.

3.1.1.1 Veri setleri

Her iki algoritma için veri seti olarak 20 Newsgroups (Anonymous. 2008) veri seti kullanılmıştır. Bu veriseti 20 farklı haber grubundan ve toplam 18846 adet dokümandan oluşmaktadır. Veriseti 2, 4 ve 8 katı ile kopyalanarak farklı büyüklükte veri setleri elde edilmiştir. Deneylerde farklı büyüklükte 4 farklı veri seti kullanılmıştır.

3.1.1.2 Bayes teoremi

Bayes teoremi şartlı olasılık üzerine kurulmuştur. Şartlı olasılık ise bir olay olduğunda diğer olayın olma olasılığıdır. Şartlı olasılık formülü aşağıda verilmiştir.

$$P(H|E) = \frac{P(E|H) \times P(H)}{P(E)} \quad (3.1)$$

Eşitlik 3.1'de:

$P(H)$: H hipotezinin olma olasılığıdır.

$P(E)$: E olayın olma olasılığıdır.

$P(E|H)$: H hipotezi doğru olduğunda E'nin olma olasılığıdır.

$P(H|E)$: E doğru olduğunda H'nin olma olasılığıdır.

$$P(H|E) = \frac{P(E|H) \times P(H)}{P(E|H) \times P(H) + P(E|-H) \times P(-H)} \quad (3.2)$$

Eşitlik 3.1 ve **eşitlik 3.2** aynıdır. **Eşitlik 3.2**'deki:

$P(-H)$: H hipotezinin doğru olmama olasılığıdır.

$P(E|-H)$ H hipotezi doğru olmadığında E'nin olma olasılığıdır.

3.1.1.3 Naif Bayes sınıflayıcı

Naif Bayes algoritması, Bayes teoremini kullanarak verilen verinin her bir sınıf için üye olma olasılığını tahmin eder. Hangi üyelik değeri olasılığı daha yüksekse, veri o sınıfa atanır. Bu sınıflama algoritmasına göre verilerin tüm özellikleri birbirinden bağımsızdır (Naif özelliği).

y sınıfı ve $x_1, \ldots\ldots, x_n$ bağımsız özellik vektörü verildiğinde (Anonymous 2019a)

$$P(y|x_1, \ldots., x_n) = \frac{P(y)P(x_1,\ldots.,x_n|y)}{P(x_1,\ldots\ldots.,x_n)} \quad (3.3)$$

$$P(y|x_1, \ldots xn) = \frac{\prod_{i=1}^{n} P(x_i|y)}{P(x_{1,\ldots.},x_n)}\ P(y) \quad (3.4)$$

$P(x_1, \ldots, x_n)$ veriye verilen bir sabit olduğundan aşağıdaki sınıflama kuralı uygulanabilir:

$$P(y|x_1, \ldots. x_n) \infty\ P(y) \prod_{i=1}^{n} P(x_i|y) \quad (3.5)$$

$$\hat{y} = argmax_{\ y}\ \ P(y) \prod_{i=1}^{n} P(x_i|y) \quad (3.6)$$

3.1.1.4 Çok terimli Naif Bayes

Metin sınıflamada kullanılan Naif Bayes çeşidi olup multinomally dağıtılmış veriler için Naif Bayes algoritmasını uygulamaktadır. Dağılım, vektörler tarafından parametrize edilmektedir (Anonymous 2019a).

$$\theta_y = (\theta_{y1}, \ldots, \theta_{yn}) \quad (3.7)$$

Eşitlik 3.7'de y sınıftır. n ise özellik sayısı olup, bu özellik sayısı metin sınıflamada kelimelerin büyüklüğüdür. θ_{yi} y sınıfında bulunan örneğin i özelliğinin Px_i olasılığıdır.

θ_y parametreleri maksimum olasılığın yumuşatılmış versiyonuyla tahmin edilmektedir (Anonymous 2019a).

$$\hat{\theta_{yi}} = \frac{N_{yi}+\alpha}{N_y+\alpha n} \tag{3.8}$$

Eşitlik 3.8'de $N_{yi} = \sum_{x \in T} x_i$ T eğitim setindeki y sınıfında bulunan i özelliğinin bulunma sayısıdır. $N_y = \sum_{i=1}^{|T|} N_{yi}$ ise y sınıfındaki tüm özelliklerin sayısıdır. α ise yumuşatma parametresi olup eğitim setinde olmayan bir özelliğin de bir sınıfa atanmasını sağlamakta ve sıfır olasılık durumunu önlemektedir. $\alpha = 1$ seçilmesi Laplace yumuşatma, $\alpha < 1$ seçilmesi ise Lidstone yumuşatma olarak adlandırılmaktadır. Bu çalışmada ise Lidstone yumuşatma kullanılmıştır (Anonymous 2019a).

3.1.1.5 Tamamlayıcı Naif Bayes

Tamamlayıcı Naif Bayes algoritması, standart çok terimli Naif Bayes algoritmasının bir uyarlamasıdır. Naif Bayes sınıflama algoritması, dengesiz eğitim setlerinde doğru sonuçlar verememektedir. Örneğin; bir sınıfa ait olan örnek sayısı diğer sınıflara oranla daha fazlaysa, Naif Bayes algoritması sınıf atamasını örnek sayısı fazla olan sınıfa eğilimli olarak yapmaktadır. Bunu önlemek için Tamamlayıcı Naif Bayes sınıflama algoritması kullanılmaktadır. Naif Bayes algoritması ağırlık tahmini için sadece tek bir sınıftaki eğitim setini kullanırken, Tamamlayıcı Naif Bayes algoritması o sınıf dışındaki tüm sınıflarda bulunan eğitim örneklerini kullanmaktadır.

3.1.1.6 Hadoop

Hadoop, açık kaynak kodlu bir yazılım çerçevesidir. Ölçeklenebilir, güvenilir ve dağıtık bir hesaplama sistemi için geliştirilmiştir. Büyük verinin hız, hacim ve çeşitlilik gibi özelikleri sonucunda süregelen sorunlarını çözebilmektedir.

Hadoop kendi servisi sayesinde kümede bulunan herhangi bir düğüm arızası durumunu tespit edebilmekte ve bu düğümde bulunan veriyi diğer makinelere dağıtmaktadır. Böylece yüksek hata toleransı özelliğine sahip olan pahalı sunucuların alımına gerek kalmamaktadır.

Hadoop iki önemli elementten oluşmaktadır: MapReduce ve Hadoop Dağıtılmış Dosya Sistemi (HDFS).

MapReduce elementi "map" yani "harita" ve "reduce" yani "azaltmak" olmak üzere iki aşamadan oluşmaktadır. Map aşamasında veri anahtar/değer çiftlerine çevrilmekte, "reduce" aşamasında ise veri toplanmakta, üzerinde hesaplamalar yapılmakta ve veri tek bir değere dönüştürülmektedir. İşlemler verinin bulunduğu bloğa en yakın düğümde gerçekleştirildiğinden girdi/çıktı maliyeti azalmakta, böylece hız artmaktadır (Zheng 2014).

Hadoop'u oluşturan ikinci eleman Dağıtılmış Dosya Sistemi (HDFS)'dir. Veri setleri HDFS'ye yüklendiğinde, veri 64MB-128 MB büyüklüğünde olan bloklara dağıtılacaktır ve dağıtılan veriler kopyalanarak diğer düğümlerde bulunan bloklardaki yerini alacaktır. Böylece bir düğümde sorun olduğunda veri kaybolmayacak, kopya olan veriden devam edilecektir. HDFS, bir adet ad düğümü (NameNode) ve birçok veri düğümünden (DataNode) oluşmaktadır. NameNode, veri ve dosya sistemi hakkındaki bilgileri tutmaktadır. Örneğin, bir veri hangi DataNode 'da ve bu Datanode'un hangi bloğunda, kaç adet kopyası var, kopyalar hangi bloklarda gibi bilgileri tutmakta ve yönetimi sağlamaktadır. Örneğin, bir blokta ya da DataNode'da sorun olduğunda, NameNode bunu tespit edebilmekte ve aynı görev için diğer DataNode'ları ve blokları görevlendirmektedir. DataNode'lar veriyi bloklar halinde tutmaktadır ve tüm işlemler DataNode'lar üzerinde olmaktadır. Sonuç olarak kullanılabilirlik ve hata toleransı artmaktadır (Zheng 2014).

3.1.1.7 Performans değerlendirme kriterleri

Bu çalışmada performans değerlendirme kriterleri olarak doğruluk, hassasiyet, hatırlama, F ölçüsü, ortalama güvenilirlik, kappa ölçümleri tercih edilmiştir. Bu ölçümlerin hesaplanmasında sınıflama probleminde doğru ve yanlış tahminlerin özetini gösteren karışıklık matrisi kullanılmıştır. İki sınıf için karışıklık matrisi **çizelge 3.1**'de verilmiştir. **Çizelge 3.1**'de de görüldüğü üzere Pozitif sınıfına ait olan bir örnek Pozitif olarak sınıflandırıldığında DP yani Doğru Pozitif olmakta; Negatif olarak sınıflandırıldığında ise YN yani Yanlış Negatif olmaktadır. Negatif sınıfındaki bir örnek Pozitif sınıfına konduğunda YP yani Yanlış Pozitif, doğru sınıfa yani Negatif sınıfına konduğunda DN yani Doğru Negatif olmaktadır.

Çizelge 3.1 İki sınıf için karışıklık matrisi

		Tahmin Edilen Sınıf	
		Pozitif	**Negatif**
Gerçek Sınıf	**Pozitif**	Doğru Pozitif (DP)	Yanlış Negatif (YN)
	Negatif	Yanlış Pozitif (YP)	Doğru Negatif (DN)

Doğruluk: Formülü **eşitlik 3.9**'da verilmiştir. (Saito ve Rehmsmeier 2015). **Eşitlik 3.9**'da da görüldüğü üzere doğruluk değeri Doğru Negatif ve Doğru Pozitif toplamının, Doğru Negatif, Yanlış Pozitif, Yanlış Negatif ve Doğru Pozitif toplamına oranıdır.

$$\text{Doğruluk} = \frac{DN+DP}{DN+YP+YN+DP} \tag{3.9}$$

Hatırlama: Formülü **eşitlik 3.10**'da verilmiştir (Saito ve Rehmsmeier 2015). **Eşitlik 3.10**'da da verildiği üzere hatırlama değeri Doğru Pozitif değerinin Yanlış Negatif ve Doğru Pozitif değer toplamına oranıdır.

$$Hatırlama = \frac{DP}{YN+DP} \tag{3.10}$$

Hassasiyet: Formülü **eşitlik 3.11**'de verilmiştir (Saito ve Rehmsmeier 2015). Hassasiyet değeri Doğru Pozitif değerinin Yanlış Pozitif ve Doğru Pozitif toplamına oranıdır.

$$Hassasiyet = \frac{DP}{YP+DP} \tag{3.11}$$

F- ölçüsü: Formülü **eşitlik 3.12**'de verilmiştir (Saito ve Rehmsmeier 2015). F-ölçüsü değeri, Hatırlama ve Hassasiyet değerlerinin çarpımının 2 katının Hatırlama ve Hassasiyet toplamına oranıdır.

$$F \text{ ölçüsü} = \frac{2xHatırlamaxHassasiyet}{Hatırlama+Hassasiyet} \tag{3.12}$$

Kappa: Formülü **eşitlik 3.13**'te verilmiştir. Bu eşitliğe göre OA gözlenen anlaşma iken, AC tesadüfen yapılan anlaşmadır. OA değeri iki sınıflandırıcının rastgele seçilen veri noktasını sınıflarken ikisinin de doğru ya da yanlış olma olasılığıdır. AC ise iki sınıflandırıcının rastgele seçilen veri noktası üzerinde tesadüfen anlaşmış olma olasılığıdır (Kuncheva 2013).

$$Kappa = \frac{OA-AC}{1-AC} \tag{3.13}$$

Doğru sınıflanan örnek yüzdesi: Doğru olarak sınıflanan örnek sayısının yüzde değerinin hesaplanmasıyla elde edilen değerdir.

Eğitim süresi: Modelin eğitilmesi için gerekli olan süredir.

Test süresi: Modelin test edilmesi için gerekli olan süredir.

3.1.1.8 Ağırlık ve normalizasyon yöntemleri

Bu çalışmada Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları her biri için performansı artırmak adına ağırlık yöntemleri olarak terim frekansı (tf) ve terim frekansı-ters belge frekansı (tfidf); normalizasyon yöntemleri olarak l1 (n1) ve l2 (lnorm) yöntemleri kullanılmıştır. Çalışmada kullanılan ağırlık ve normalizasyon yöntemleri bu bölümde formülleriyle açıklanmıştır.

Terim frekansı: Bir terimin belge içerisinde hangi sıklıkta bulunduğunu ölçmektedir. **Eşitlik 3.14**'te formulize edilmiştir. Terim frekansı, t teriminin belge içerisindeki bulunma sayısının belgedeki tüm terimlerin sayısına oranıdır.

$$Terim\ frekansı(t) = \frac{t\ teriminin\ belge\ içerisindeki\ bulunma\ sayısı}{Belgedeki\ tüm\ terimlerin\ sayısı} \quad (3.14)$$

Ters belge frekansı (idf): Terimin ne derece önemli olduğunu ölçmektedir. TF değerini hesaplarken, bütün terimlerin önemi eşit olarak alınmıştır. Ama, "ve", "sonra", "ile" gibi bazı terimler az bir öneme sahip olmasına rağmen belge içerisinde sıklıkla bulunabilmektedir. Bu durumda, çok sıklıkla kullanılan terimlerin ağırlığını azaltırken, az sayıda kullanılanları artırmamız gerekmektedir. Ters belge frekansı **eşitlik 3.15**'te formülize edilmiştir. Ters belge frekansı, toplam belge sayısının e tabanında log değerinin, t terimini içeren belge sayısına oranıdır.

$$Ters\ belge\ frekansı(t) = \frac{\log_e Toplam\ belge\ sayısı}{t\ terimini\ içeren\ belge\ sayısı} \quad (3.15)$$

Terim frekansı-ters belge frekansı (TF-IDF): Terim Frekansı ve Ters Belge Frekansı değerlerinin çarpımıdır. TF-IDF değeri metin sınıflandırmada kullanılabilmekte ve ilgi analizi yapılabilmektedir.

Normalizasyon ile iki doküman arasındaki kelimelerin sıklığı ölçülmekte olup toplam kelime sayısının etkisi kaldırılmaktadır. Örneğin; iki belgeden daha uzun olanı daha fazla kelimeye sahip olacağından farklı birer uzunluğa sahip olacak ve farklı görüneceklerdir. Yalnız, farklılık konusunda belgenin uzunluğu yerine anlamı daha önemlidir. Belgelerin uzunluğunun etkisini kaldırmak adına normalizasyon işlemi yapılmaktadır.

Bu çalışmada normalizasyon yöntemi olarak lnorm ve n1 normalizasyon yöntemleri kullanılmıştır. Lnorm normalizasyon yöntemi "L2 norm" ya da "Öklid uzunluğu" olarak da adlandırılabilmekte olup vektörler arasındaki mesafeyi hesaplarken Öklid uzaklığı kullanırken, n1 normalizasyonu "Manhattan mesafesi" kullanmaktadır.

3.1.2 Duygu analizi

Çalışmanın bu kısmında bu deney için kullanılan veri setleri ve performans değerlendirme kriterleri hakkında bilgi verilmiştir. Bu çalışmada tezin 4.1 bölümünde açıklanan Naif Bayes, Tamamlayıcı Naif Bayes algoritmaları ve normalizasyon, ağırlık ve yumuşatma parametreleri kullanılarak Eclipse platformunda mahout kütüphanesi aracılığı ile Java programlama dili kullanılarak duygu analizinde bulunulmuştur. Modelin oluşturulması için tezin 4.1 bölümünde verilen ve en iyi doğruluk performansını gösteren parametreler kullanılmıştır. Her bir veri seti için bu parametreler kullanılarak eğitim sağlanmış, ardından test edilmiştir. Performans değerlendirme kriteri olarak doğruluk, hassasiyet, hatırlama ve F-ölçüsü kullanılmıştır. İşlemci teknolojisi olarak Core i7, işlemci çekirdek sayısı 4, işlemci hızı 2.0 GHz olan bir bilgisayarda deney yapılmıştır.

3.1.2.1 Veri setleri

Duygu analizi için 3 farklı veri seti kullanılmıştır. Bu veri setleri, pozitif ve negatif kelimelerden oluşan veri setleri (Hu ve Liu 2004) ve "Amazon movie reviews (Anonymous 2015a)" veri setidir.

Duygu analizinin gerçekleştirilmesi için pozitif ve negatif olan kelimeler elde edilmiştir. Pozitif ve negatif kelimeleri içeren veri setleri birleştirilmiştir. Oluşan veri setinde 2005 adet pozitif kelime, 4783 adet negatif kelime bulunmuştur. Pozitif kelimeler 1 sınıfı ile, negatif kelimeler 0 sınıfı ile ifade edilmiştir.

Bu bölümde test amaçlı kullanılan veri seti "Amazon movie review" (Anonymous 2015a) veri setidir. Bu veri seti yaklaşık 8 milyon adet yorumdan oluşmaktadır. Veri seti "productId", "userId", "profileName", "score", "time", "summary", "text" gibi kısımlardan oluşmaktadır. Bu kısımlardan "text" kısmı film hakkında yapılan yorumları temsil etmektedir.

3.1.2.2 Performans değerlendirme kriterleri

Performans değerlendirme kriteri olarak doğruluk, hassasiyet, hatırlama ve F-ölçüsü kullanılmıştır. Bu kriterler bölüm 3.1.1.7'de açıklanmıştır.

3.1.3 Öneri sistemi geliştirilmesi

Çalışmanın bu kısmında deney için kullanılan veri setleri, mesafe ölçüleri, performans değerlendirme kriterleri hakkında bilgiler verilmiştir. Bu çalışmada mahout kütüphaneleri yardımıyla, işlemci teknolojisi olarak Core i7, işlemci çekirdek sayısı 4, işlemci hızı 2.0 GHz olan bir bilgisayar kullanılarak Eclipse platformunda kullanıcı tabanlı bir öneri sistemi geliştirilmiştir ve geliştirilen öneri sisteminin performansı performans değerlendirme kriterleri ile değerlendirilmiştir. Veri seti olarak MovieLens 100K veri setinin tamamı ile Movielens 20M verisetinin bir bölümü kullanılmıştır. Kullanıcılar arasındaki benzerliği hesaplamak için Pearson, Öklid, ortalanmamış kosinüs, log olasılık ölçümleri kullanılırken; performans değerlendirme kriterleri olarak ortalama mutlak hata, kök ortalama karesel hata, hassasiyet, hatırlama ve f-ölçütü gibi ölçütler kullanılmıştır.

3.1.3.1 Veri setleri

Bu çalışmada Movielens 100K veri setinin tümü ve Movielens 20M veri setinin (Anonymous 2013) bir bölümü kullanılmıştır. Movielens 100K veri seti 1700 film, 1000 kullanıcı ve 100,000 oydan oluşmaktadır. Movielens 20M ise 20 milyon oydan oluşmaktadır. Bu çalışmada ise 20M verisetinin rastgele 1 milyonu kullanılmıştır.

3.1.3.2 Mesafe ölçüleri

Bu çalışmada mesafe ölçüsü olarak Pearson korelasyonu, kosinüs benzerlik, Öklid uzaklık ölçütü, log benzerlik ve merkezlenmemiş kosinüs benzerliği kullanılmıştır.

3.1.3.2.1 Pearson korelasyonu

Pearson korelasyonu iki değişken arasındaki doğrusal ilişkinin yönünü ve derecesi ölçmek için kullanılan korelasyonlardan biridir. **Eşitlik 3.16**'da formülü verilmiştir (Gong 2008).

$$sim\ (i,j) = \frac{\sum_{c \in I_{i,j}} (R_{i,c} - A_i)(R_{j,c} - A_j)}{\sqrt{\sum_{c \in I_{i,j}} (R_{i,c} - A_i)^2 * \sum_{c \in Iij} (R_{j,c} - A_j)^2}} \quad (3.16)$$

Bu eşitlikte $R_{i,c}$ değeri *i* kullanıcısının *c* ürününe verdiği oydur. A_i ise i kullanıcısının birlikte derecelendirilen tüm ürünler için ortalama oyudur. I_{ij} değeri *i* ve *j* kullanıcıları tarafından ortak olarak oylanan ürünler kümesidir. R_{jc} ise *j* kullanıcısının *c* ürününe verdiği oydur. A_j ise *j* kullanıcısının birlikte derecelendirilen tüm ürünler için ortalama oyudur.

3.1.3.2.2 Kosinüs benzerlik

Veri madenciliğinde en çok kullanılan benzerlik hesaplama yöntemlerinden biridir. Metinler arasındaki benzerlik vektörel olarak ölçülmektedir.

Eşitlik 3.17'de formülü verilmiştir (Gong 2008).

$$sim\ (i,j) = \frac{\sum_{k=1}^{n} R_{i,k} * R_{j,k}}{\sqrt{\sum_{k=1}^{n} {R_{i,k}}^2 * {R_{j,k}}^2}} \tag{3.17}$$

Eşitlik 3.17'de $R_{i,k}$ değeri *i* kullanıcısı tarafından *k* ürününe verilen oydur. $R_{j,k}$ ise *j* kullanıcısının *k* ürününe verdiği oydur. *n* ise her iki kullanıcı tarafından oylanan ürünlerin sayısıdır.

3.1.3.2.3 Öklid uzaklık ölçütü

Öklid uzaklık ölçütü, iki obje arasındaki doğrusal uzaklıktır ve benzerlik hesaplamada kullanılabilmektedir.

Eşitlik 3.18'te formülü verilmiştir (Siddiquee vd. 2014).

$$D_{uv} = \sqrt{\sum_{i=1}^{n} |a_{ui} - a_{vi}|^2} \tag{3.18}$$

Eşitlik 3.18'te *U* tüm kullanıcılar kümesini temsil ederken, D_{uv} ise *u* ve *v* kullanıcıları arasındaki verdikleri film oyları bazlı mesafedir. a_{ui} ve a_{vi} değerleri ise *u* ve *v* kullanıcılarının *i* ürünü için verdikleri oydur.

3.1.3.2.4 Log benzerlik

Eğer *y* dizisi (p_i,q_i) gibi Fourier katsayıları değerlerine sahipse *y*'ye ait olan periodogram $a_i = {p_i}^2 + {q_i}^2$ eşitliğine eşittir. Log benzerlik, **eşitlik 3.19**'da verilmiştir (Bagnall vd. 2005).

$$l(a) = \frac{\sum_{i=1}^{n-1} a_i}{2a_i} \log(2a_i) \tag{3.19}$$

3.1.3.2.5 Merkezlenmemiş kosinüs benzerliği

Merkezlenmemiş kosinüs benzerliği, iki tercih vektörü arasındaki açının cosinüs değerini ölçmektedir ve verileri ortalamamaktadır (Anonymous 2017b).

3.1.3.3 Performans değerlendirme kriterleri

Bu bölümde performans değerlendirme kriteri olarak kök ortalama karesel hata, ortalama mutlak hata, hassasiyet, hatırlama ve F-ölçüsü kriterleri kullanılmıştır. Kullanılan kriterlerden kök ortalama karesel hata ve ortalama mutlak hata kriterleri bu bölümde açıklanırken, hassasiyet, hatırlama ve F-ölçüsü kriterleri bölüm 3.1.1.7'de açıklanmıştır.

Kök ortalama karesel hata: Kök ortalama karesel hata, karesel hataların ortalamasından sonra kökünün alınmasıyla hesaplanmaktadır. **Eşitlik 3.20**'de formülü verilmiştir (Birtolo vd. 2011).

$$Kök\ Ortalama\ Karesel\ Hata = \sqrt{\frac{1}{N} * \sum_{u,i}(p_i(u) - r_i(u))^2} \tag{3.20}$$

Eşitlik 3.20'te *N* değeri toplam oy sayısını, $p_i(u)$ değeri *i* ürünü için *u* kullanıcısı tarafından verilecek olan oy tahminini, $r_i(u)$ ise gerçek oyu temsil etmektedir. Daha küçük kök ortalama kare hata değerleri, daha iyi performans anlamına gelmektedir.

Ortalama mutlak hata: Ortalama mutlak hata, mutlak hataların aritmetik ortalamasının alınmasıyla hesaplanmaktadır. Bu hata ölçütünün formülü **eşitlik 3.21**'de verilmiştir.

$$Ortalama\ Mutlak\ Hata = \frac{1}{N} * \sum_{u,i}(p_i(u) - r_i(u)) \quad (3.21)$$

Eşitlik 3.21'de N değeri toplam oy sayısını, $p_i(u)$ değeri i ürünü için u kullanıcısı tarafından verilecek olan oy tahminini, $r_i(u)$ ise gerçek oyu temsil etmektedir.

Hassasiyet, hatırlama ve f-ölçüsü: Bu kriterler Bölüm 3.1.1.7'de açıklanmıştır.

3.1.4 Öneri sistemlerinin tahmin performansını iyileştirmek için evrimsel sinir ağları

Çalışmanın bu kısmında bu deney için kullanılan veri setleri, algoritmalar, performans değerlendirme kriterleri hakkında bilgiler verilmiştir.

Bu çalışmada öneri sistemleri yapay sinir ağları ve genetik algoritmalar kullanılarak iyileştirilmiştir. Ardından, iyileştirilen öneri sistemleri performansı literatürde yapılan çalışmalarla karşılaştırılmıştır. Ortalama karesel hata ve kök ortalama karesel hata değerleri performans değerlendirme kriteri olarak kullanılmıştır. İşlemci teknolojisi olarak Core i7, işlemci çekirdek sayısı 4, işlemci hızı 2.0 GHz olan bir bilgisayarda java programlama diliyle deneyler yapılmıştır. Sonuç olarak iyileştirilmiş öneri sistemlerinin performansının literatürde bulunan çalışmalardan daha iyi olduğu gözlenmiştir.

Bu çalışmada öneri sistemlerinin tahmin performansı, yapay sinir ağlarının tahmin performansı ve yapay sinir ağlarının genetik algoritmalar ile birlikte kullanılması ile oluşan öneri sisteminin tahmin performansıyla karşılaştırılmıştır. Yapay Sinir Ağları olarak çok katmanlı algılayıcı, genelleştirilmiş ileri beslemeli ağ ve koaktif sinirsel bulanık çıkarım sistemi kullanılmıştır.

3.1.4.1 Veri seti

Movielens 100K ve Movielens 1M verisetleri bu çalışmada kullanılmıştır (Anonymous 2013). Movielens 100K veriseti 1000 adet kullanıcı ve 1700 adet film içeren yaklaşık 100,000 adet oydan oluşmaktadır. Movielens 1M veriseti ise 6000 kullanıcı ve 4000 film içeren yaklaşık 1 milyon adet oydan oluşmaktadır. Bu çalışmada her iki veri seti için veri setinin %10'u rastgele olarak çapraz doğrulama için kullanılırken, %20'si test verisi olarak kullanılmıştır.

3.1.4.2 Algoritmalar

Bu bölümde çok katmanlı algılayıcı, genelleştirilmiş ileri beslemeli ağ ve koaktif sinirsel bulanık çıkarım sistemi kullanılmıştır.

3.1.4.2.1 Çok katmanlı algılayıcı

Çok katmanlı algılayıcı, ileri beslemeli sinir ağlarıdır. Girdi katmanı ve çıktı katmanı arasında bir ya da birden fazla gizli katman bulunabilmektedir ve her düğümde bulunan doğrusal olmayan aktivasyon fonksiyonu sayesinde karmaşık karar fonksiyonları oluşturabilmektedir (Guha ve Patra 2010, Gibson vd. 1989, Meyer ve Pfeiffer 1993).

Çok katmanı algılayıcıda ağa verilen *s(k)* girdisine göre ağın ürettiği *y(k)* çıktısı **eşitlik 3.22**'de verilmiştir (Guha ve Patra 2010).

$$y(k) = f\left(\sum_{j=0}^{n-1}\left(s_j w_j\right) + b\right) \tag{3.22}$$

Eşitlik 3.22'de b bias, w_j ise j girdisi ile ilişkilendirilmiş ağırlık matrisidir.

3.1.4.2.2 Genelleştirilmiş ileri besleme

Genelleştirilmiş ileri besleme algoritması, çok katmanlı algılayıcının genelleştirilmiş hali olup, bağlantılar bir ya da birden fazla katmandan atlayabilmektedir (Jadhav vd. 2012).

Aynı sayıda işleme elemanı içeren çok katmanlı algılayıcı ve genelleştirilmiş ileri besleme yöntemlerinden çok katmanlı algılayıcı eğitim için daha fazla devir değerine ihtiyaç duymaktadır. Genelleştirilmiş ileri beslenme yöntemi eğitim için genel olarak geri yayılım algoritmasını kullanmaktadır. Bu algoritmayla hata değerleri ağ üzerinde yayılmakta ve gizli işleme elemanlarının adaptasyonu sağlanmaktadır (Jadhav vd. 2012).

3.1.4.2.3 Koaktif sinirsel bulanık çıkarım sistemi

Koaktif sinirsel bulanık çıkarım sistemi (CANFIS), sinir ağına bulanık girdilerin entegrasyonu ile oluşan ağlardır (Hemachandra ve Satyanarayana 2013). Bu sistemde ilk katmanda bulunan herbir düğüm $x_1, x_2, \ldots, x_n$ ile ifade edilmekte olup bulanık kümenin üyelik derecesini göstermektedir. İlk katmana ait çıktıların çarpımı hesaplanmakta ve ikinci katmana iletilmektedir. Ardından ağırlık normalizasyonu hesaplanmaktadır (Chiroma vd. 2013).

3.1.4.2.4 Genetik algoritma

Genetik algoritma popülasyon tabanlı bir optimizasyon metodu olup doğal seleksiyon ve türlerin adaptasyon kapasitesine dayanmaktadır (Barbosa ve Gouvea 2012). Genetik algoritmanın herbir basamağında popülasyon içinden rastgele bireyler seçilmekte, bu bireyler yeni bireylerin oluşumunda kullanılmakta, sonuç olarak popülasyon sürekli güncellenmektedir. Genetik algoritma kullanımında 3 ana kural vardır. Bu kurallar seçim, çaprazlama ve mutasyon olarak sıralanabilmektedir. Bu kurallardan seçim kuralı popülasyondan ebeveyn olarak da adlandırılabilinen bireylerin seçimi olup, bu bireyler yeni nesilin oluşumunda kullanılacaktır. Çaprazlama kuralı yeni bireyin oluşması için iki

bireyi birleştirmektedir. Mutasyon kuralında ise yeni bireyin oluşması için ebeveyn bireylere rastgele değişiklikler uygulanmaktadır (Anonymous 2020c).

Biyolojide, popülasyondaki herbir konu genlerden oluşan kromozomlar tarafından temsil edilmektedir. Herbir kromozom uygunluk değerine göre değerlendirilmekte, daha iyi uygunluk değerine sahip olan kromozomların yeni bireylerin oluşumunda kullanılma ihtimali artmaktadır (Kuhn ve Johnson 2019). Ardından, çaprazlama ve mutasyon uygulanmakta, yeni bireyler oluşturulmaktadır. Seçme, çaprazlama ve mutasyon işlemleri belirlenen sayıda nesil oluşuncaya kadar ya da sonlandırma kriteri sağlanana kadar devam etmektedir (Maulik ve Bandyopadhyay 2000).

3.1.4.3 Öğrenme kuralları

Bu çalışmada öğrenme kuralı olarak eşlenik gradyan ve momentum yöntemleri kullanılmıştır.

Eşlenik gradyan: Hem yinelemeli algoritma hem de doğrudan yöntem olarak kullanılan bu yöntem sayısal bir çözüm üretmekte olup doğrusal ve doğrusal olmayan sistemleri optimize etmekte kullanılabilmektedir. Bu yöntem, büyük sistemlerin çözülmesini sağlamaktadır (Anonymous 2015c).

Momentum: Bu algoritma gradyan inişinin hızın artırabilmekte, yüksek eğrilik, küçük ama tutarlı eğim gibi durumlarda öğrenmeyi daha kısa bir sürede gerçekleştirmektedir. (Anonymous 2020d).

3.1.4.4 Bulanık modeller

Yapay sinir ağı olarak koaktif sinirsel bulanık çıkarım sistemi kullanıldığında bulanık model olarak Takagi-Sugeno-Kang ve Tsukamoto bulanık modelleri sistemin eğitiminde ve testinde kullanılmıştır.

Takagi-Sugeno-Kang: Bu modelin giriş kısımları girdi değişkenlerinin doğrusal fonksiyonları ile temsil edilmekte olup karmaşık doğrusal olmayan sistemlere yaklaşabilmektedir. Bunu sağlamak için birkaç kural kullanmasına rağmen daha yüksek modelleme doğruluğu sağlamaktadır (Kwak vd. 2015).

Tsukamoto: Bu model, ikili olmayan ve doğrusal olmayan problemleri çözmek için kullanılabilmektedir. Sonuca ağırlıklı ortalama hesaplanarak ulaşılmaktadır (Sayuti ve Sofyan 2018).

3.1.4.5 Performans değerlendirme kriterleri

Bu çalışmada performans değerlendirme kriterleri olarak ortalama karesel hata ve kök ortalama karesel hata değerleri kullanılmıştır. Kök ortalama karesel hata formülü bölüm 3.1.3.3'te verilmiştir. Ortalama kare hatası ise, kök ortalama kare hatası formülünde kullanılan kök ifadesinin kaldırılmasıyla elde edilmektedir.

3.2 Yöntem

Bu bölümde tez çalışmasında yapılan tüm deneyler için kullanılan yöntemler açıklanmıştır.

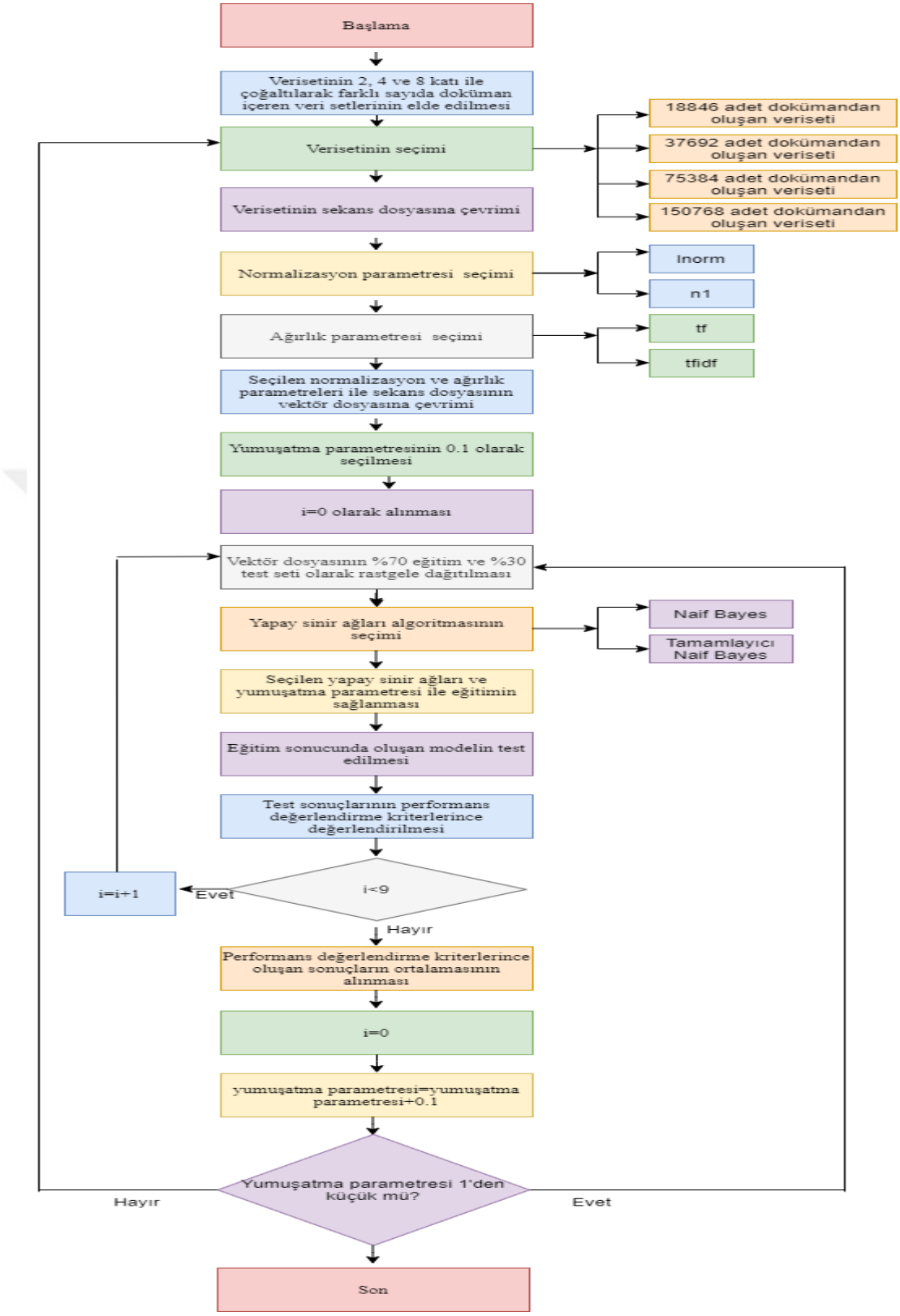
3.2.1 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile büyük veri analizi

Uygulanan yönteme ait olan akış şeması **şekil 3.1**'de verilmiştir. **Şekil 3.1**'e göre veri seti 2, 4 ve 8 katı ile kopyalanarak farklı büyüklükte veri setleri elde edilmiştir. Ardından, oluşan veri setlerinden bir veri seti seçilmiş, bu veri seti sekans dosyasına çevrilmiştir. Oluşan sekans dosyasının seçilen ağırlık ve normalizasyon parametresi ile vektör dosyasına çevrilmesi ile deneye devam etmiştir. Yumuşatma parametresi başlangıç değeri olarak 0.1 olarak belirlenmiştir. Vektör dosyasının rastgele %70'i eğitim seti olarak seçilirken, geriye kalan %30'u test seti olarak belirlenmiştir. Seçilen yapay sinir ağı ve yumuşatma parametresi ile eğitim sağlanmıştır, ardından test edilmiştir. Test sonuçları performans değerlendirme kriterlerince değerlendirilmiştir. Ardından, vektör dosyasının

rastgele %70 eğitim ve %30 test seti olarak dağıtılması kısmına geri dönülmüş, yeni eğitim ve test setleriyle oluşan modelin performansı değerlendirilmiştir. Farklı eğitim setleriyle ve aynı yumuşatma parametresi ile modelin eğitilmesi ve farklı test setleriyle modelin test edilip performans değerlendirmesi 10 kez tekrar edilmiştir. Ardından, oluşan performans sonuçlarının ortalaması alınmıştır. Daha sonra yumuşatma parametresine 0.1 değeri eklenerek artırılması sağlanmış, oluşan yumuşatma parametresi değeri 1'den küçükse, deney vektör dosyasının rastgele eğitim ve test seti olarak dağıtılması kısmına geri dönmüştür. Böylece deney herbir yumuşatma parametresi için vektör dosyasının rastgele dağıtılmasıyla oluşan farklı veri setleriyle eğitilmesi, test edilmesi ve deneyin bu kısmının 10 kez tekrarlanması ile oluşan sonuçların ortalamasının alınmasıyla devam etmiştir. Oluşan yumuşatma parametresi 1 değerinden küçük değilse deney veri setinin seçimi kısmına dönerek en başından başlanmıştır.

Deneyler tüm veri setleri (18846 adet doküman, 37692 adet doküman, 75384 adet doküman, 150768 adet doküman), tüm normalizasyon ve ağırlık tipi kombinasyonları (lnorm-tf, lnorm-tfidf, n1-tf, n1-tfidf), tüm yumuşatma parametresi değerleri [0.1,0.9] ve tüm yapay sinir ağları algoritmaları (Naif Bayes, Tamamlayıcı Naif Bayes) için Mahout kütüphanesi yardımıyla ayrı ayrı yapılmıştır. Performans değerlendirme kriteri olarak doğruluk, kappa, ağırlıklı hatırlama, ağırlıklı hassasiyet, ortalama güvenilirlik, ağırlıklı F1 derecesi, ortalama doğru sınıflanan örnek yüzdesi, test süresi ve eğitim için gerekli olan eğitim süresi kullanılmıştır. Bu kriterlerin ortalaması alınarak nihai sonuca ulaşılmıştır.

Sonuç olarak, değişen normalizasyon, yumuşatma ve ağırlık parametrelerine karşı değişen Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları performansı incelenmiş, her bir veri seti için seçilen performans değelendirme kriterine göre en iyi performansı gösteren parametreler (normalizasyon, ağırlık ve yumuşatma parametreleri) tespit edilmiştir. Bu iki algoritma için gerekli olan eğitim ve test süreleri incelenmiştir.



Şekil 3.1 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları performans karşılaştırımını içeren deneyin akış şeması

3.2.2 Duygu Analizi

Pozitif ve negatif kelimelerden oluşan veri seti 2, 4 ve 8 katı ile çoğaltılmıştır. Her bir veri seti için bölüm 4.1'de verilen parametrelerden doğruluk performans değerlendirme kriterine göre en iyi performansı gösteren parametreler kullanılarak (Naif Bayes algoritması gerçek veri seti için lnorm-tfidf ve yumuşatma parametresi=0.8, verinin iki katı ile çoğaltılması ile oluşan veriseti için lnorm-tfidf ve yumaşatma parametresi=0.1, verinin 4 katı ile çoğaltılması ile oluşan veriseti için lnorm-tfidf ve yumuşatma parametresi=0.1, verinin 8 katı ile çoğaltılması sonucu oluşan veriseti için lnorm-tf ve yumuşatma parametresi=0.1 kullanılırken; Tamamlayıcı Naif Bayes algoritması gerçek veriseti için n1-tfidf ve yumuşatma parametresi=0.1, verinin iki katı ile çoğaltılması ile oluşan veriseti için lnorm-tfidf ve yumaşatma parametresi=0.1, verinin 4 katı ile çoğaltılması ile oluşan veriseti için lnorm-tf ve yumuşatma parametresi=0.3, verinin 8 katı ile çoğaltılması sonucu oluşan veriseti için lnorm-tf ve yumuşatma parametresi=0.1) her bir veri seti için Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile eğitim sağlanmıştır. Oluşturulan modellerle "Amazon movie review" veri setindeki yorumları içeren "text" kısmı Eclipse platformunda mahout kütüphanesi yardımıyla pozitif ya da negatif olarak sınıflandırılmıştır (Anonymous 2015b). Herbir pozitif ve negatif sınıfı için bir skor hesaplanmıştır. Eğer bu skorlar arasındaki fark Naif Bayes algoritması için 10'dan küçükse, Tamamlayıcı Naif Bayes algoritması için 0.001'den küçükse bu yorum "nötr" olarak sınıflandırılmıştır. Böylece "Amazon movie review" veri setine yeni bir özellik eklenmiştir.

Oluşturulan sistemin performans değerlendirmesi şu şekilde yapılmıştır. Orijinal "Amazon movie review" veri setindeki 1 ve 2 olarak oylanan oyların negatif sınıfına; 3 olarak oylanan oyların nötr sınıfına; 4 ve 5 olarak oylanan oyların pozitif sınıfına ait oldukları kabul edilmiştir. Bu sınıflama gerçek sınıfları temsil etmiştir. Ardından, Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile oluşturulan model sayesinde sınıflanan "Amazon movie review" verisetindeki veriler, gerçek "Amazon movie review" verisetindeki veriler aracılığıyla performans değerlendirme kriterlerince değerlendirilmiştir. En doğru doğruluk değerini veren sonucun oluşturduğu veri seti ileriki deneylerde kullanılmıştır.

Oluşan yeni verisetindeki 1 ve 2 olarak oylanan ve nötr sınıfına konan veriler negatif sınıfına atanmıştır. 3, 4 ve 5 olarak oylanan ve nötr sınıfına konan veriler pozitif sınıfına konmuştur. Ardından lojistik regresyon sınıflama algoritması ile verilen oylar ve duygular arasındaki ilişki gözlenmiştir. Veri setinin rastgele %30'u test seti için ayrılmış, 3 farklı eğitim ve test setleri elde edilmiştir. İşlemler her bir veri seti değeri için 3'er kez yapılmış, sonuçların ortalaması alınmıştır. Her bir deneyde lambda değeri 0.0001 olarak alınmıştır. Deneyler, öğrenme kat sayısı 0.001, 0.0001 ve 0.00001 için 3 farklı veri seti ile 3'er kez yapılmış, sonuçların ortalaması alınmıştır.

3.2.3 Öneri sistemi geliştirilmesi

Bu çalışmada mahout kütüphaneleri yardımıyla, Eclipse platformunda kullanıcı tabanlı bir öneri sistemi geliştirilmiştir ve geliştirilen öneri sisteminin performansı mutlak hata, kök ortalama karesel hata, hassasiyet, hatırlama ve f-ölçütü gibi değerlendirme kriterleri ile değerlendirilmiştir. Veri seti olarak MovieLens 100K veri setinin tamamı ile Movielens 20M verisetinin bir bölümü kullanılmıştır. Veri setinin rastgele %70'i eğitim setini oluştururken, %30'u test setini oluşturmuştur. Kullanıcılar arasındaki benzerliği hesaplamak için Pearson, Öklid, ortalanmamış kosinüs, log olasılık ölçümleri kullanılmıştır. Her bir deney, herbir benzerlik ölçütü için 3'er kez yapılmış, sonuçların ortalaması alınmıştır.

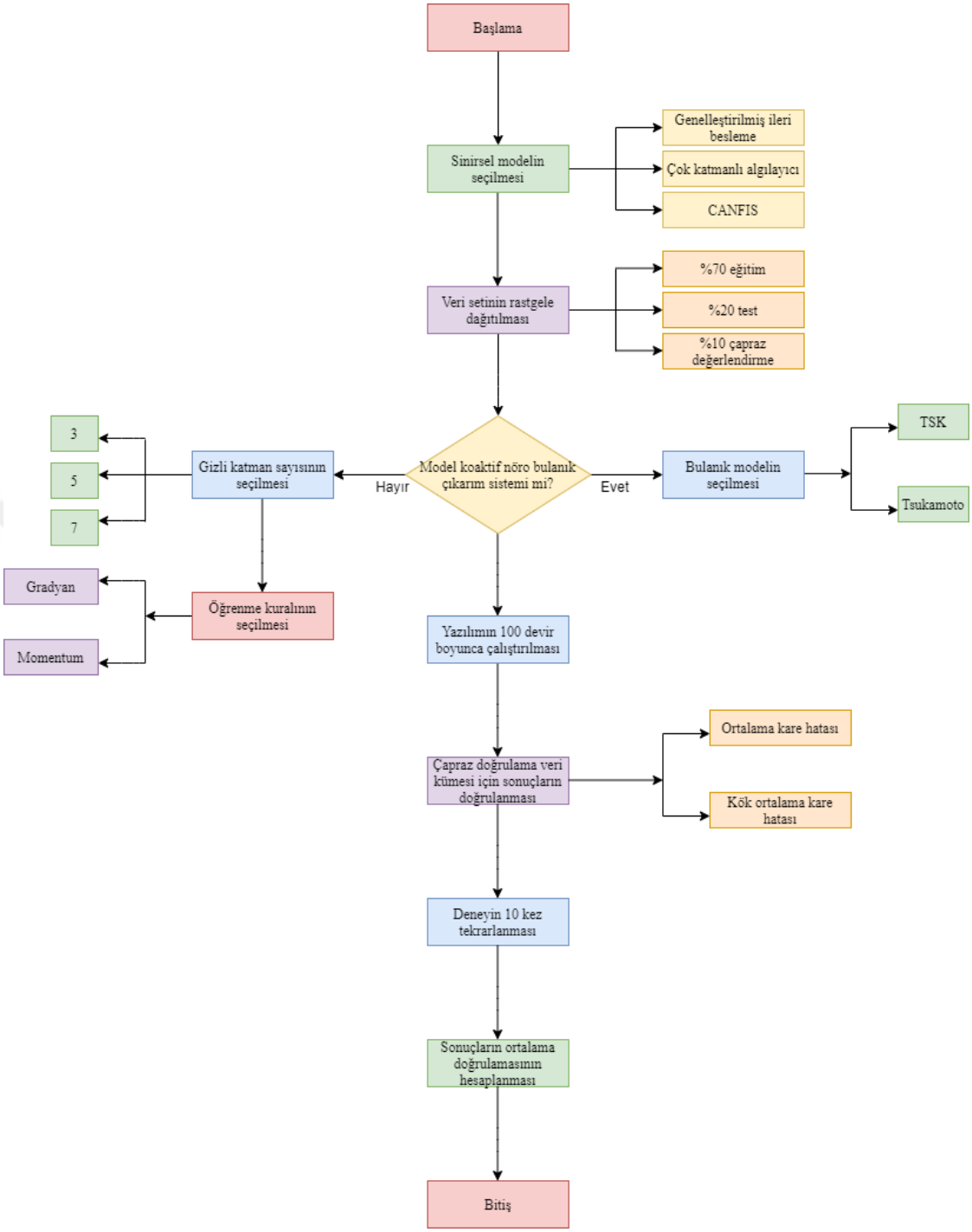
3.2.4 Öneri sistemlerinin tahmin performansını iyileştirmek için evrimsel sinir ağları

Bu çalışmada öneri sistemlerinin tahmin performansı yapay sinir ağlarının tahmin performansı ve yapay sinir ağlarının genetik algoritmalar ile birlikte kullanılması ile oluşan öneri sisteminin tahmin performansıyla karşılaştırılmıştır. Yapay Sinir Ağları olarak çok katmanlı algılayıcı, genelleştirilmiş ileri besleme ve koaktif nöro bulanık çıkarım sisteminden yararlanılmıştır. Sistemi eğitmek ve test etmek için Movielens 100K ve Movielens 1M veri setleri kullanılmıştır. Veri setinin rastgele %10'u çapraz doğrulama veri seti olarak ayrılırken, %20'si test seti olarak ayrılmıştır. Çok katmanlı algılayıcı ve genelleştirilmiş ileri besleme yapay sinir ağları için deneyler 3, 5 ve 7 gizli katman için ayrı

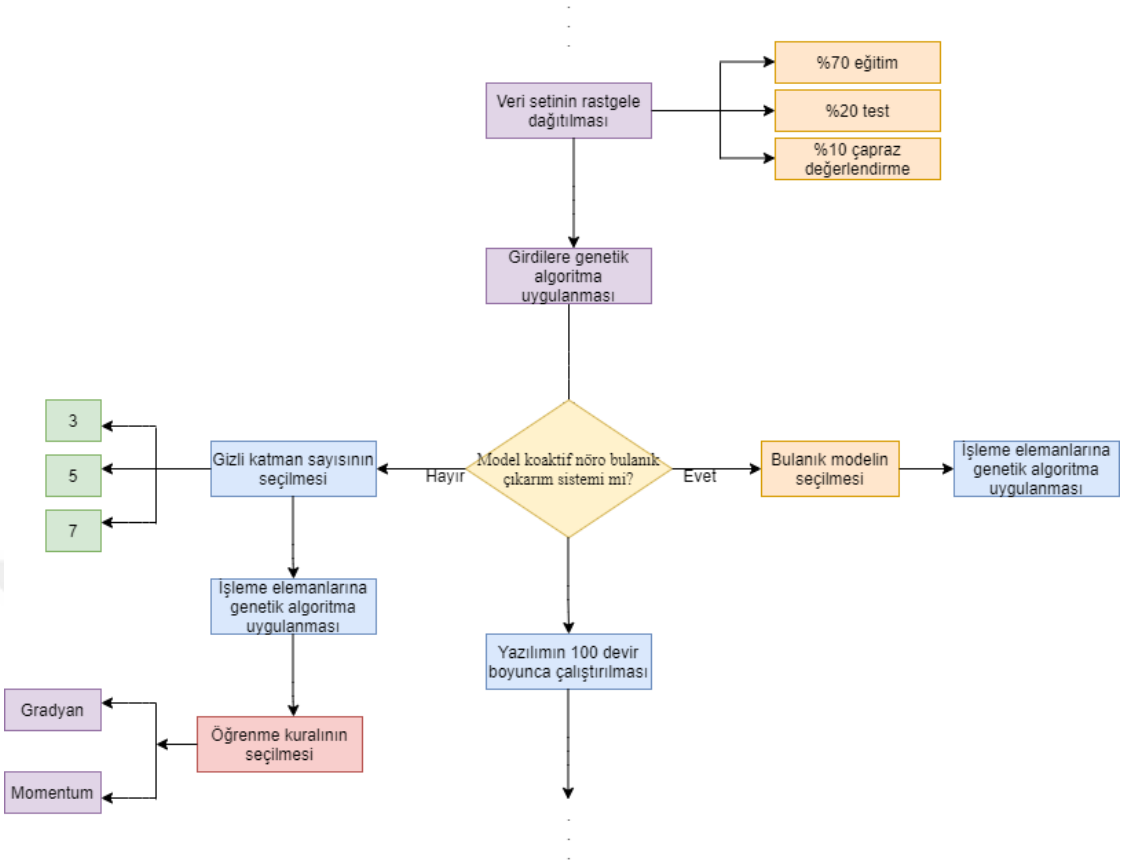
ayrı yapılmıştır. Tüm yapay sinir ağları çeşitleri için momentum ve eşlenik gradyan öğrenme kuralları daha iyi bir sonuç almak için denenmiştir. Koaktif nöro bulanık çıkarım sistemi modeli için Bell fonksiyonu üyelik fonksiyonu olarak kullanılırken, TSK ve Tsukamoto modelleri fuzzy modeli olarak kullanılmıştır. Ortalama karesel hata ve kök ortalama karesel hata kriterleri performans değerlendirme kriteri olarak seçilmiştir.

Girdilerin model için önem derecesi bilinmediğinden, girdilere genetik algoritma uygulanmıştır. Sonuç olarak, genetik algoritma ile model için en düşük hata oranını verecek olan girdi kombinasyonları belirlenmiştir. Buna ek olarak, gizli katmandaki işleme elemanlarının sayısını optimize etmek amacıyla işleme elemanlarına da genetik algoritma uygulanmıştır.

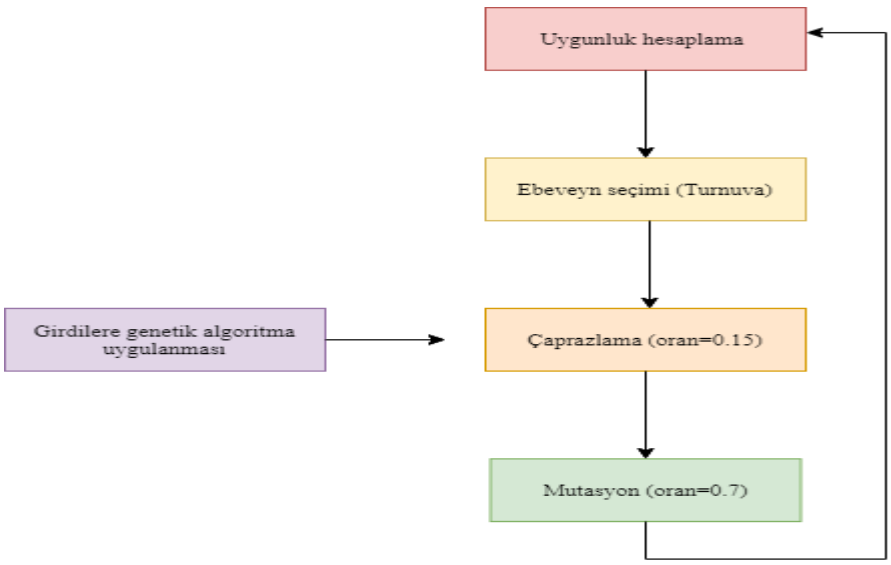
Tüm deneyler 10 kez tekrar edilmiş, çapraz doğrulama veri seti için ortalama karesel hata ve kök ortalama karesel hata değerleri alınmıştır. Akış şemaları **şekil 3.2** ve **şekil 3.3** ve **şekil 3.4**'te verilmiştir.



Şekil 3.2 Yapay sinir ağları ile oluşan öneri sistemine ait olan akış şeması



Şekil 3.3 Evrimsel sinir ağı tabanlı tahmin sisteminin akış şeması



Şekil 3.4 Evrimsel sinir ağları tabanlı tahmin sistemine ait olan genetik algoritma detayları

4. ARAŞTIRMA BULGULARI

Bu bölümde tez çalışmasında yapılan tüm deneylere ait olan araştırma bulguları ve sonuçlar yer almaktadır.

4.1 Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları İle Büyük Veri Analizi

Bu bölümde Naif Bayes ve Tamamlayıcı Naif Bayes algoritması ile veri setlerine ve performans değerlendirme kriterlerine göre en iyi performansı gösteren normalizasyon, ağırlık, yumuşatma parametreleri hakkında bilgi verilmiştir. Değişen yumuşatma, normalizasyon, ağırlık parametreleri ve veri büyüklüğüne göre, Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının ortalama doğruluk, ortalama ağırlıklı hatırlama, ortalama ağırlıklı hassasiyet ve ortalama doğru sınıflanan örnek yüzdesi, eğitim ve test süreleri detaylı analizi yapılmıştır ve T testi ve F testi ile Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları performansları karşılaştırılmıştır.

4.1.1 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile kullanılan veri setleri, ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

Naif Bayes algoritması ile en iyi performansı gösteren yumuşatma parametreleri her bir veri seti için **çizelge 4.1, 4.2, 4.3, 4.4**'te verilmiştir. Yumuşatma parametresi α ile ifade edilmiştir. **Çizelge 4.1** ve **çizelge 4.2**'de görüldüğü üzere tüm performans değerlendirme kriterleri açısından en iyi performansı gösteren normalizasyon ve ağırlık parametresi lnorm-tfidf olmuştur. **Çizelge 4.1**'e göre tüm performans değerlendirme kriterleri baz alındığında genel olarak en iyi yumuşatma parametresi 0.8 olurken, bu durum **çizelge 4.2** için 0.1 olmuştur. **Çizelge 4.3**'e göre tüm performans değerlendirme kriterleri bazında genel olarak en iyi performans lnorm-tfidf normalizasyon ve ağırlık parametresi ve 0.1 yumuşatma parametresiyle elde edilirken, **çizelge 4.4**'e göre bu durum lnorm-tf normalizasyon ve ağırlık parametresi ve 0.1 yumuşatma parametresi ile elde edilmiştir.

Çizelge 4.1 Naif Bayes Algoritması ile 18846 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.8	**91.14689**	0.1	89.00329	0.1	83.97805	0.2	86.78322
Ortalama kappa	0.5	**0.87916**	0.1	0.85786	0.1	0.80118	0.1	0.83198
Ortalama doğruluk	0.8	**91.14689**	0.1	89.00329	0.1	83.9780	0.2	86.7832
Ortalama güvenilirlik	0.6	**86.88716**	0.1	83.6314	0.1	78.3253	0.1	62.20272
Ortalama ağırlıklı hassasiyet	0.8	**0.91563**	0.1	0.89994	0.1	0.8612	0.4	0.88989
Ortalama ağırlıklı hatırlama	0.8	**0.91147**	0.1	0.89005	0.1	0.83978	0.2	0.86783
Ortalama ağırlıklı F1 derecesi	0.8	**0.9108**	0.1	0.88595	0.1	0.83145	0.1	0.85789

Çizelge 4.2 Naif Bayes algoritması ile 37692 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.1	**96.976**	0.1	94.1226	0.1	90.56828	0.1	93.64076
Ortalama kappa	0.1	**0.95901**	0.1	0.926	0.1	0.88453	0.1	0.92274
Ortalama doğruluk	0.1	**96.94823**	0.1	94.1226	0.1	90.5682	0.1	93.64076
Ortalama güvenilirlik	0.1	**92.14597**	0.1	87.6737	0.1	83.66365	0.1	89.11959
Ortalama ağırlıklı hassasiyet	0.1	**0.97068**	0.1	0.9454	0.1	0.91313	0.1	0.94462
Ortalama ağırlıklı hatırlama	0.1	**0.96948**	0.1	0.9412	0.1	0.90567	0.1	0.93641
Ortalama ağırlıklı F1 derecesi	0.3	**0.96956**	0.1	0.9361	0.1	0.89726	0.1	0.92929

Çizelge 4.3 Naif Bayes algoritması ile 75384 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.4	**99.82282**	0.1	97.26648	0.1	93.94926	0.1 ve 0.2	99.79185
Ortalama kappa	0.1	**0.99604**	0.1	0.9646	0.1	0.92605	0.1	0.9931
Ortalama doğruluk	0.1	**99.82282**	0.1	97.26648	0.1	93.94926	0.1 ve 0.2	99.79185
Ortalama güvenilirlik	0.1	**89.83651**	0.1	85.70741	0.1	80.49726	0.2	71.9843
Ortalama ağırlıklı hassasiyet	0.1	0.99824	0.1	0.97415	0.1	0.94548	[0.1,0.9]	**1**
Ortalama ağırlıklı hatırlama	0.1	**0.99824**	0.1	0.97267	0.1	0.93951	0.1 ve 0.2	0.99791
Ortalama ağırlıklı F1 derecesi	0.1	0.99824	0.1	0.97069	0.1	0.92845	0.1 ve 0.2	**0.9996**

Çizelge 4.4 Naif Bayes algoritması ile 150768 adet dokümandan oluşan veri seti kullanılan ağırlık ve normalizasyon parametrelerince için en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.1	99.1639	0.1	98.98704	0.1	97.69051	0.1	**99.58967**
Ortalama kappa	0.1	0.98832	0.1	0.98552	0.1	0.96889	0.1	**0.98891**
Ortalama doğruluk	0.1	99.16398	0.1	98.9870	0.1	97.69051	0.1	**99.5896**
Ortalama güvenilirlik	0.1	**82.66703**	0.1	82.5250	0.1	81.4832	0.1	75.16584
Ortalama ağırlık hassasiyet	0.1	0.99165	0.1	0.98991	0.1	0.97699	[0.1,0.8]	**1**
Ortalama ağırlıklı hatırlama	0.1	0.99165	0.1	0.98986	0.1	0.9769	0.1	**0.9959**
Ortalama ağırlıklı F1 derecesi	0.1	0.99159	0.1	0.98987	0.1	0.97691	0.1	**0.99796**

Tamamlayıcı Naif Bayes algoritması ile en iyi performansı gösteren yumuşatma parametreleri her bir veri seti için **çizelge 4.5, 4.6, 4.7** ve **4.8**'de verilmiştir. **Çizelge 4.5**'e

göre tüm performans değerlendirme kriterleri baz alındığında en iyi performans 0.1 yumuşatma parametresi ve n1-tfidf normalizasyon ve ağırlık parametreleriyle elde edilmiştir. **Çizelge 4.6**'ya göre en iyi performans lnorm-tfidf normalizasyon ve ağırlık parametreleriyle ve genel olarak 0.1 yumuşatma parametresi ile oluşurken, **çizelge 4.7**'de bu durum genel olarak lnorm-tf normalizasyon ve ağırlık parametresi ve 0.3 yumuşatma parametresiyle oluşmuştur. **Çizelge 4.8**'de de genel olarak en iyi performans lnorm-tf normalizasyon ve ağırlık parametresi tarafından gösterilirken, yumuşatma parametresi genel olarak değeri 0.1 değerini almıştır.

Çizelge 4.5 Tamamlayıcı Naif Bayes algoritması ile 18846 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.9	89.50852	0.1	**91.49856**	0.1	89.79198	0.9	81.38633
Ortalama kappa	0.9	0.85608	0.1	**0.88348**	0.1	0.86452	0.9	0.74073
Ortalama doğruluk	0.9	89.50	0.1	**91.49856**	0.1	89.79198	0.9	81.38633
Ortalama güvenilirlik	0.9	85.05672	0.1	**86.44161**	0.1	84.47463	0.3	63.48922
Ortalama ağırlıklı hassasiyet	0.9	0.89649	0.1	**0.91677**	0.1	0.9016	0.9	0.90703
Ortalama ağırlıklı hatırlama	0.9	0.89509	0.1	**0.91499**	0.1	0.89792	0.9	0.81386
Ortalama ağırlıklı F1 derecesi	0.9	0.89294	0.1	**0.91298**	0.1	0.89444	0.9	0.85005

Çizelge 4.6 Tamamlayıcı Naif Bayes algoritması ile 37692 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.1	**96.91629**	0.2	96.33747	0.1	94.77751	0.9	94.34443
Ortalama kappa	0.4	**0.95864**	0.1	0.95041	0.1	0.93203	0.2	0.93597

Çizelge 4.6 Tamamlayıcı Naif Bayes algoritması ile 37692 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri (devam)

Ortalama doğruluk	0.1	**96.91629**	0.1	96.33747	0.1	94.77751	0.9	94.34443
Ortalama güvenilirlik	0.1	**92.14289**	0.1	90.98914	0.1	89.07228	0.3	89.62566
Ortalama ağırlıklı hassasiyeti	0.1	**0.97045**	0.1	0.96377	0.1	0.94883	0.3	0.94679
Ortalama ağırlıklı hatırlama	0.1	**0.96917**	0.1	0.96338	0.1	0.94778	0.9	0.94344
Ortalama ağırlıklı F1 derecesi	0.1	**0.96899**	0.1	0.96303	0.1	0.94661	0.9	0.94071

Çizelge 4.7 Tamamlayıcı Naif Bayes algoritması ile 75384 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.4	99.75162	0.1	98.91801	0.1	97.72794	0.3	**99.89593**
Ortalama kappa	0.4	**0.99461**	0.1	0.98437	0.1	0.97024	0.3	0.99304
Ortalama doğruluk	0.5	99.75162	0.1	98.91801	0.1	97.72794	0.3	**99.89593**
Ortalama güvenilirlik	0.8	**89.7981**	0.1	88.53099	0.1	86.66268	0.3	59.98126
Ortalama ağırlıklı hassasiyet	0.8	0.99751	0.1	0.98931	0.1	0.97798	[0.1,0.9]	**1**
Ortalama ağırlıklı hatırlama	0.8	0.99732	0.1	0.98293	0.1	0.97729	0.3	**0.99895**
Ortalama ağırlıklı F1 derecesi	0.8	0.99751	0.1	0.98903	0.1	0.97609	0.3	**0.99949**

Çizelge 4.8 Tamamlayıcı Naif Bayes algoritması ile 150768 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri

	Lnorm-Tfidf		**N1-Tfidf**		**N1-Tf**		**Lnorm-Tf**	
	α	**Değer**	**α**	**Değer**	**α**	**Değer**	**α**	**Değer**
Ortalama doğru sınıflanan örnek yüzdesi	0.9	81.9466	0.1	99.22381	0.1	98.35264	0.1	**99.59278**

Çizelge 4.8 Tamamlayıcı Naif Bayes algoritması ile 150768 adet dokümandan oluşan veri seti için kullanılan ağırlık ve normalizasyon parametrelerince en iyi performansı gösteren yumuşatma parametreleri (devam)

Ortalama kappa	0.9	0.77483	0.1	0.98854	0.1	0.97734	0.1	**0.9891**
Ortalama doğruluk	0.9	82.08307	0.1	99.2238	0.1	98.35264	0.1	**99.59278**
Ortalama güvenilirlik	0.7	69.09853	0.1	**82.7129**	0.1	82.01792	0.1	73.16544
Ortalama ağırlık hassasiyet	0.1	0.86689	0.1	0.99225	0.1	0.98355	[0.1,0.9]	**1**
Ortalama ağırlıklı hatırlama	0.9	0.82083	0.1	0.99225	0.1	0.98354	0.1	**0.99592**
Ortalama ağırlıklı F1 derecesi	0.9	0.7677	0.1	0.99225	0.1	0.9835	0.1	**0.99796**

4.1.2 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile veri setlerine ve performans değerlendirme kriterlerine göre en iyi performansı gösteren parametreler ve değerleri

Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları ile veri setlerine ve performans değerlendirme kriterlerine göre en iyi performası gösteren parametreler ve değerleri **çizelge 4.9** ve **çizelge 4.10**'da verilmiştir. Değerler iki hane olarak ifade edilmiştir. **Çizelge 4.9** ve **çizelge 4.10**'da normalizasyon *n* ile, ağırlık *a* ile, yumuşatma parametresi α ile, oluşan performans değeri ise *d* ile gösterilmiştir.

Çizelge 4.9'a göre Naif Bayes algoritması kullanıldığında 18846 adet dokümandan oluşan veri seti için tüm performans değerleri baz alındığında genel olarak en iyi performans yumuşatma parametresi **0.8**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tfidf** seçildiğinde oluşurken; 37692 adet dokümandan oluşan veri seti için bu değer yumuşatma parametresi **0.1**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tfidf** seçildiğinde; 75384 adet dokümandan oluşan veri seti için yumuşatma parametresi **0.1**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tfidf** seçildiğinde; 150768 adet dokümandan oluşan veri seti için yumuşatma parametresi **0.1**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tf** seçildiğinde oluşmuştur. Naif Bayes algoritması tüm performans değerleri

baz alındığında en iyi performası genel olarak yumuşatma parametresi olarak **0.1**, normalizasyon parametresi olarak **lnorm** kullanıldığında göstermiştir.

Çizelge 4.10'a göre Tamamlayıcı Naif Bayes algoritması için 18846 adet dokümandan oluşan veri seti için tüm performans değerleri baz alındığında genel olarak en iyi performans yumuşatma parametresi **0.1**, normalizasyon parametresi **n1**, ağırlık parametresi **tfidf** seçildiğinde gözlenirken; 37692 adet dokümandan oluşan veri seti için yumuşatma parametresi **0.1**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tfidf** seçildiğinde; 75384 adet dokümandan oluşan veri seti için yumuşatma parametresi **0.3**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tf** seçildiğinde; 150768 adet dokümandan oluşan veri seti için yumuşatma parametresi **0.1**, normalizasyon parametresi **lnorm**, ağırlık parametresi **tf** seçildiğinde gözlenmiştir. Tamamlayıcı Naif Bayes algoritması tüm performans değerleri baz alındığında en iyi performası genel olarak yumuşatma parametresi olarak **0.1**, normalizasyon parametresi olarak **lnorm** kullanıldığında göstermiştir.

Çizelge 4.9 Naif Bayes algoritması ile veri setlerine ve performans değerlendirme kriterlerine göre en iyi performansı gösteren parametreler ve değerleri

	Veri seti: 18846				**Veri seti: 37692**				**Veri Seti:75384**				**Veri Seti:150768**			
	n	**a**	**α**	**d**	**n**	**a**	**α**	**d**	**n**	**a**	**α**	**d**	**n**	**a**	**α**	**d**
Ortalama doğru sınıflanan örnek yüzdesi	lnorm	tfidf	0.8	91.15	lnorm	tfidf	0.1	96.98	lnorm	tfidf	0.4	99.82	lnorm	tf	0.1	99.59
Ortalama kappa	lnorm	tfidf	0.5	0.88	lnorm	tfidf	0.1	0.96	lnorm	tfidf	0.1	1	lnorm	tf	0.1	0.99
Ortalama doğruluk	lnorm	tfidf	0.8	91.15	lnorm	tfidf	0.1	96.95	lnorm	tfidf	0.1	99.82	lnorm	tf	0.1	99.59
Ortalama güvenilirlik	lnorm	tfidf	0.6	86.89	lnorm	tfidf	0.1	92.15	lnorm	tfidf	0.1	89.84	lnorm	tfidf	0.1	82.67
Ortalama ağırlıklı hassasiyet	lnorm	tfidf	0.8	0.92	lnorm	tfidf	0.1	0.97	lnorm	tf	[0.1,0.9]	1	lnorm	tf	[0.1,0.8]	1
Ortalama ağırlıklı hatırlama	lnorm	tfidf	0.8	0.92	lnorm	tfidf	0.1	0.967	lnorm	tfidf	0.1	1	lnorm	tf	0.1	1
Ortalama ağırlıklı F1 derecesi	lnorm	tfidf	0.8	0.91	lnorm	tfidf	0.3	0.97	lnorm	tf	0.1ve 0.2	1	lnorm	tf	0.1	1

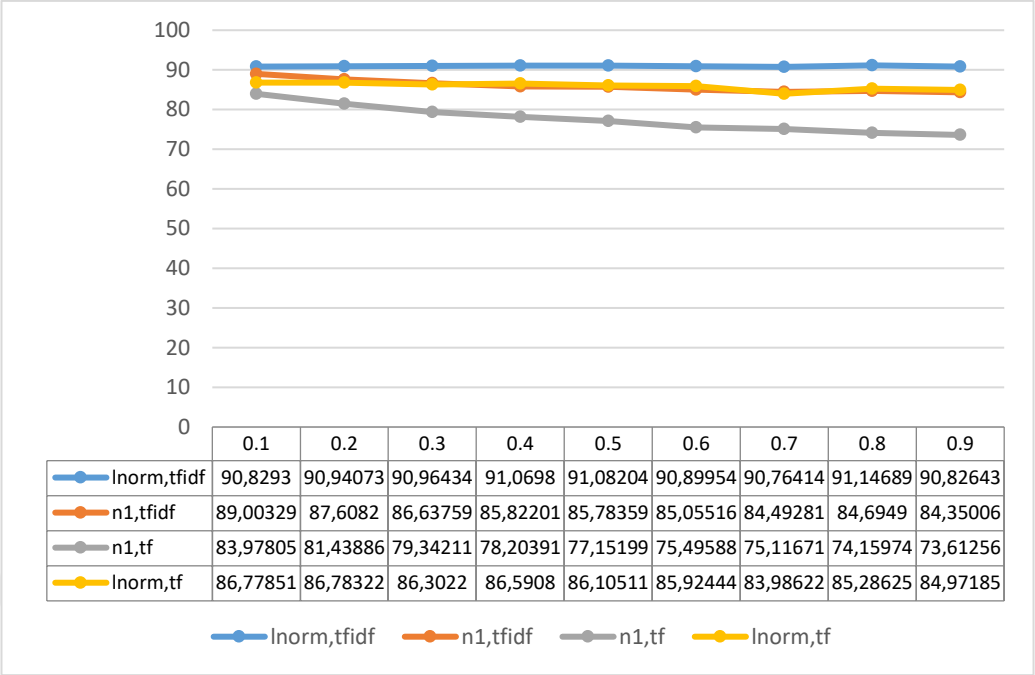
Çizelge 4.10 Tamamlayıcı Naif Bayes algoritması ile veri setlerine ve performans değerlendirme kriterlerine göre en iyi performansı gösteren parametreler ve değerleri

	Veri seti: 18846				**Veri seti: 37692**				**Veri Seti:75384**				**Veri Seti:150768**			
	n	**a**	**α**	**d**	**n**	**a**	**α**	**d**	**n**	**a**	**α**	**d**	**n**	**a**	**α**	**d**
Ortalama doğru sınıflanan örnek yüzdesi	N1	tfidf	0.1	91.50	lnorm	tfidf	0.1	96.92	lnorm	tf	0.3	99.90	lnorm	tf	0.1	99.59
Ortalama kappa	N1	tfidf	0.1	0.88	lnorm	tfidf	0.4	0.96	lnorm	tfidf	0.4	0.99	lnorm	tf	0.1	0.99
Ortalama doğruluk	N1	tfidf	0.1	91.50	lnorm	tfidf	0.1	96.92	lnorm	tf	0.3	99.90	lnorm	tf	0.1	99.59
Ortalama güvenilirlik	N1	tfidf	0.1	86.44	lnorm	tfidf	0.1	92.14	lnorm	tfidf	0.8	89.80	N1	tfidf	0.1	82.71
Ortalama ağırlıklı hassasiyet	N1	tfidf	0.1	0.92	lnorm	tfidf	0.1	0.97	lnorm	tf	[0.1,0.9]	1	lnorm	tf	[0.1,0.9]	1
Ortalama ağırlıklı hatırlama	N1	tfidf	0.1	0.91	lnorm	tfidf	0.1	0.97	lnorm	tf	0.3	1	lnorm	tf	0.1	1
Ortalama ağırlıklı F1 derecesi	N1	tfidf	0.1	0.91	lnorm	tfidf	0.1	0.97	lnorm	tf	0.3	1	lnorm	tf	0.1	1

4.1.3 Değişen yumuşatma, normalizasyon, ağırlık parametreleri ve veri büyüklüğüne göre performans incelenmesi

Bu bölümde Naif Bayes ve Tamamlayıcı Naif Bayes algoritmasına ait olan doğruluk, hatırlama ve hassasiyet performans kriterlerinin değişen yumuşatma, normalizasyon, ağırlık parametretleri ve veri büyüklüğü karşısındaki performansları analiz edilmiştir. **Şekil 4.1** ve **şekil 4.2** Naif Bayes algoritmasına ait olan performansları gösterirken, **şekil 4.3** ve **şekil 4.4** Tamamlayıcı Naif Bayes algoritmasına ait olan performansları göstermektedir. **Şekil 4.1**, **şekil 4.2**, **şekil 4.3** ve **şekil 4.4**'te *x* ekseni yumuşatma parametresini temsil ederken, *y* ekseni ortalama doğruluk değerini temsil etmektedir.

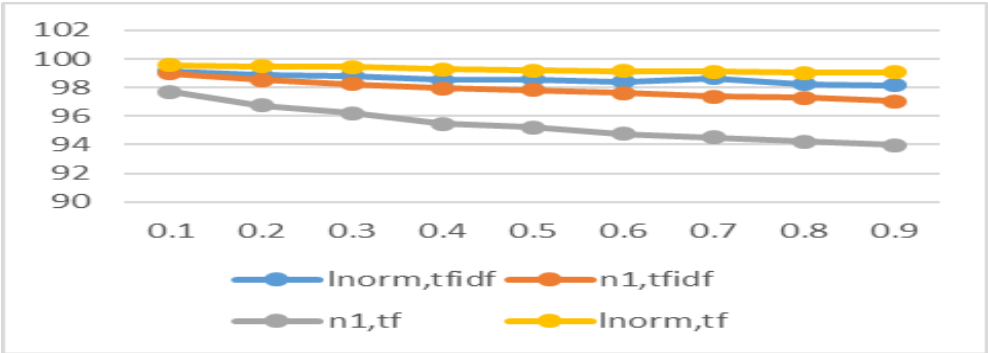
Şekil 4.1'e göre en iyi doğruluk değeri performansı 0.8 yumuşatma parametresi ile lnorm-tfidf normalizasyon ve ağırlık parametreleri kullanıldığında oluşmuştur. Diğer parametreler için, yumuşatma parametresinin artırılması genel olarak doğruluk performansını negatif yönde etkilemiştir.



	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
lnorm,tfidf	90,8293	90,94073	90,96434	91,0698	91,08204	90,89954	90,76414	91,14689	90,82643
n1,tfidf	89,00329	87,6082	86,63759	85,82201	85,78359	85,05516	84,49281	84,6949	84,35006
n1,tf	83,97805	81,43886	79,34211	78,20391	77,15199	75,49588	75,11671	74,15974	73,61256
lnorm,tf	86,77851	86,78322	86,3022	86,5908	86,10511	85,92444	83,98622	85,28625	84,97185

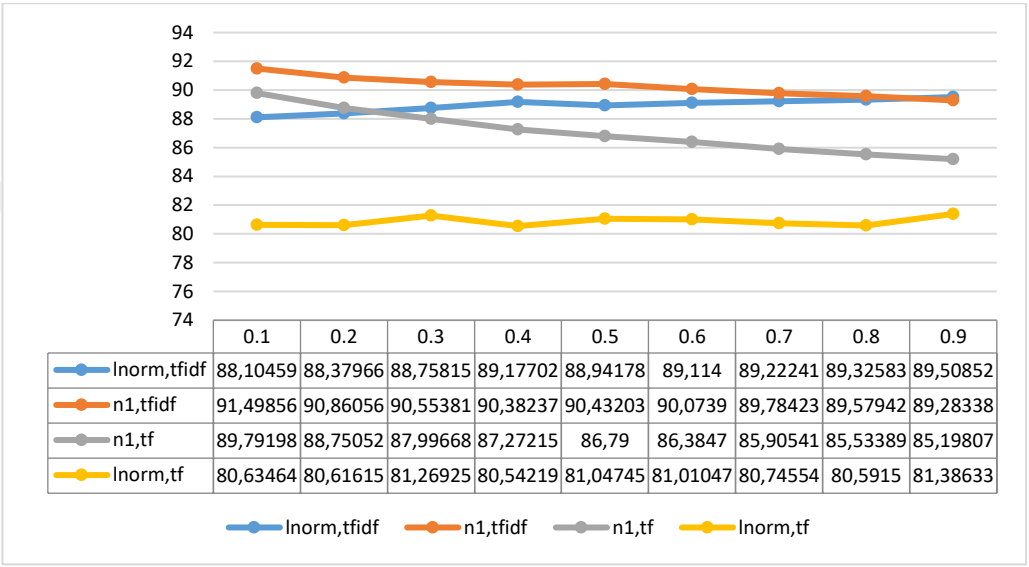
Şekil 4.1 18846 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri

Şekil 4.2'de 150768 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri verilmiştir. Bu şekle göre veri seti büyüklüğünün artması ortalama doğruluk değerini yükseltmiştir, en iyi performans değerleri tüm parametreler için yumuşatma parametresi 0.1 olarak seçildiğinde elde edilmiştir ve en iyi doğruluk değerini veren normalizasyon ve ağırlık parametresinin lnorm-tf olduğu görülmüştür.



Şekil 4.2 150768 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri

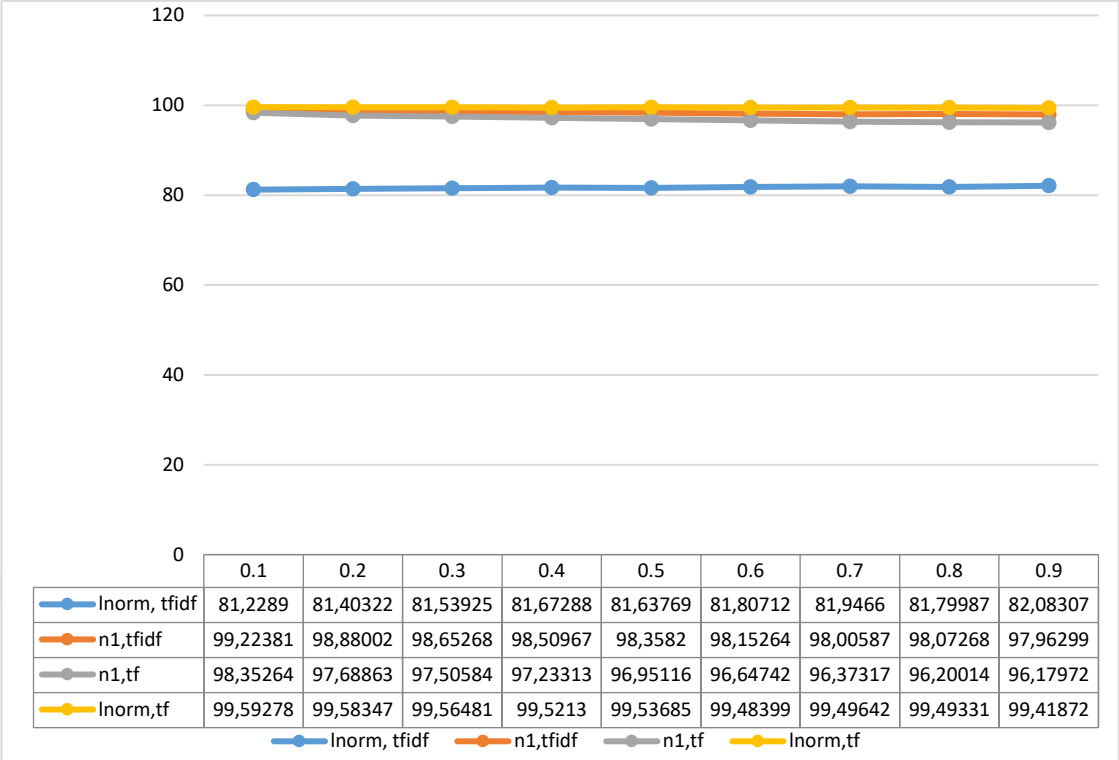
Şekil 4.3'te 18846 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Tamamlayıcı Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri verilmiştir. Bu şekle göre en iyi ortalama doğruluk değeri n1-tfidf normalizasyon ve ağırlık parametreleri ile 0.1 yumuşatma parametresi kullanıldığında oluşmuştur ve N1-tf parametreleri kullanıldığında yumuşatma parametresi arttıkça ortalama doğruluk değeri azalmıştır.



	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
lnorm,tfidf	88,10459	88,37966	88,75815	89,17702	88,94178	89,114	89,22241	89,32583	89,50852
n1,tfidf	91,49856	90,86056	90,55381	90,38237	90,43203	90,0739	89,78423	89,57942	89,28338
n1,tf	89,79198	88,75052	87,99668	87,27215	86,79	86,3847	85,90541	85,53389	85,19807
lnorm,tf	80,63464	80,61615	81,26925	80,54219	81,04745	81,01047	80,74554	80,5915	81,38633

Şekil 4.3 18846 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Tamamlayıcı Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri

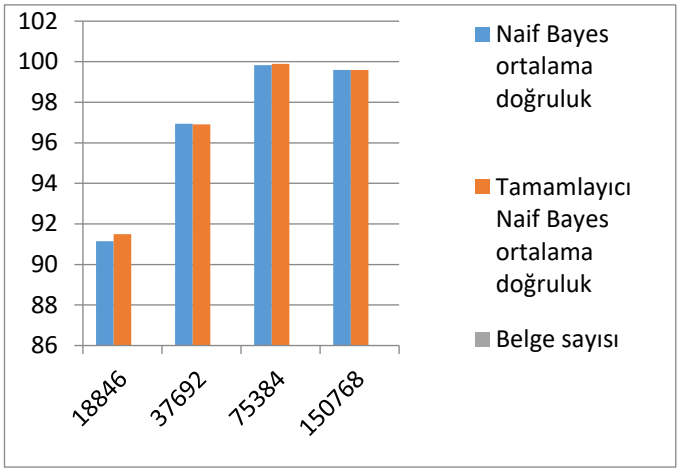
Şekil 4.4'e göre 150768 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Tamamlayıcı Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri verilmiştir. Bu şekle göre en iyi ortalama doğruluk performans değeri lnorm-tf parametreleri kullanıldığında 0.1 yumuşatma parametresi ile elde edilmektedir ve n1-tf parametreleri kullanıldığında yumuşatma parametresinin artması performansı olumsuz yönde etkilemektedir.



	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
lnorm, tfidf	81,2289	81,40322	81,53925	81,67288	81,63769	81,80712	81,9466	81,79987	82,08307
n1,tfidf	99,22381	98,88002	98,65268	98,50967	98,3582	98,15264	98,00587	98,07268	97,96299
n1,tf	98,35264	97,68863	97,50584	97,23313	96,95116	96,64742	96,37317	96,20014	96,17972
lnorm,tf	99,59278	99,58347	99,56481	99,5213	99,53685	99,48399	99,49642	99,49331	99,41872

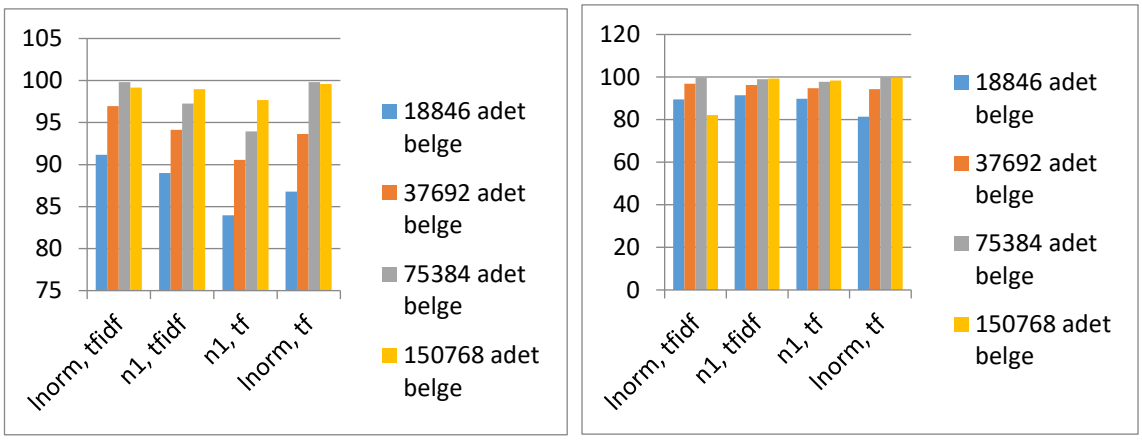
Şekil 4.4 150768 adet dokümandan oluşan veri setine ait olan, değişen normalizasyon, ağırlık ve yumuşatma parametresine göre Tamamlayıcı Naif Bayes algoritması ile oluşan ortalama doğruluk değerleri

Şekil 4.5'te her bir veri seti için Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan en iyi ortalama doğruluk değeri karşılaştırılması verilmiştir. Bu şekle göre doküman sayısı 37692 olan veri seti dışında, diğer veri setleri için en iyi ortalama doğruluk değeri Tamamlayıcı Naif Bayes algoritması kullanıldığında elde edilmiştir. Bunun nedeni, sınıflar için ağırlıklar belirlenirken, Tamamlayıcı Naif Bayes algoritmasının daha fazla örneği dikkate alarak ağırlıkları belirlemesidir.



Şekil 4.5 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan en iyi ortalama doğruluk değeri karşılaştırılması

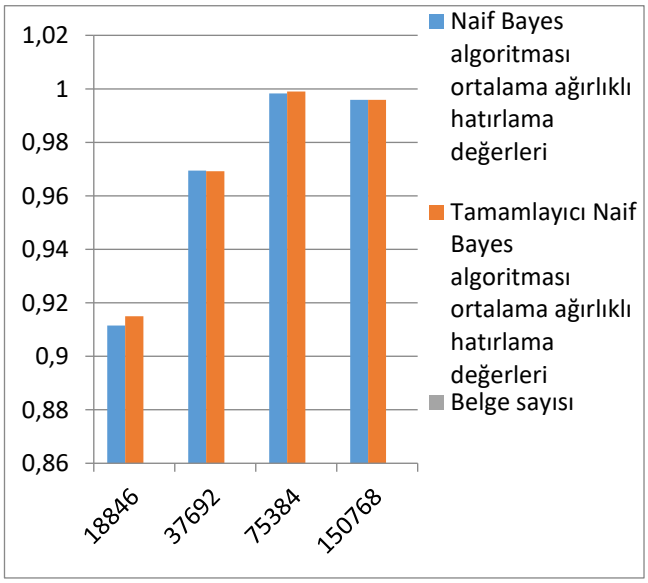
Şekil 4.6'da normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama doğruluk değerleri verilmiştir. **Şekil 4.6a**'ya göre en küçük sayıda dokümana sahip olan veri seti en iyi ortalama doğruluk performansını lnorm normalizasyon ve tfidf ağırlık parametresiyle elde etmiştir ve 37692 adet doküman ve 75384 adet doküman içeren veri setleri kullanıldığında en iyi performans lnorm-tfidf parametreleriyle elde edilirken; veri seti olarak 150768 adet doküman kullanıldığında en iyi performans lnorm-tf parametreleriyle elde edilmiştir. **Şekil 4.6b**'ye göre en küçük sayıda dokümana sahip olan veri seti en iyi ortalama doğruluk performansını n1 normalizasyon ve tfidf ağırlık parametresiyle elde etmiştir ve en iyi ortalama doğruluk değerleri 37692 adet doküman için lnorm-tfidf; 75384 ve 150768 adet doküman için lnorm-tf parametreleri ile elde edilmiştir.



(a) (b)

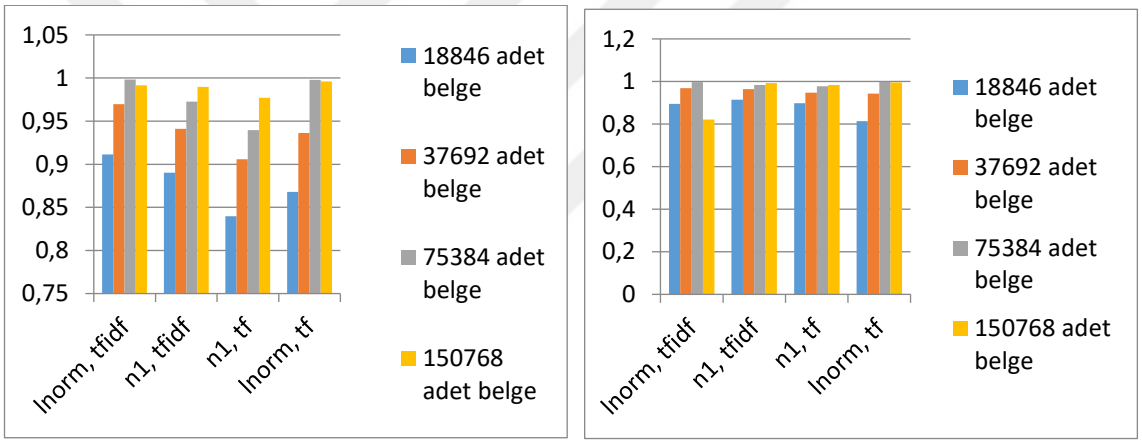
Şekil 4.6 Normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama doğruluk değerleri
(a) Naif Bayes, (b) Tamamlayıcı Naif Bayes

Şekil 4.7'de Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan ortalama ağırlıklandırılmış hatırlama değerleri karşılaştırımı verilmiştir. Bu şekle göre en iyi ortalama ağırlıklandırılmış hatırlama değeri 37692 adet dokümandan oluşan veri seti dışında diğer veri setleri için Tamamlayıcı Naif Bayes algoritmasına aittir.



Şekil 4.7 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan ortalama ağırlıklandırılmış hatırlama değerleri karşılaştırımı

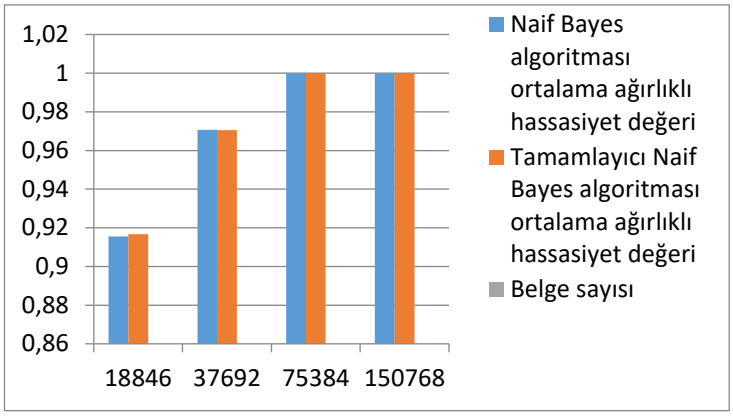
Şekil 4.8'de normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama ağırlıklandırılmış hatırlama değerleri verilmiştir. **Şekil 4.8a**'ya göre 150768 dokümandan oluşan veriseti dışında diğer veri setleri için en iyi ortalama ağırlıklandırılmış hatırlama değeri lnorm-tfidf parametreleri ile elde edilirken, 150768 adet dokümana ait en iyi hatırlama değeri lnorm-tf parametreleri ile elde edilmiştir. **Şekil 4.8b**'ye göre 18846 adet dokümandan oluşan veri seti kullanıldığında en iyi ortalama ağırlıklandırılmış hatırlama performansına n1-tfidf parametreleri ile ulaşırken; 37692 adet dokümandan oluşan veri seti kullanıldığında en iyi ortalama ağırlıklandırılmış hatırlama performansına lnorm-tfidf parametreleri ile ulaşılmıştır. 75384 ve 150768 adet dokümandan oluşan veri setleri kullanıldığında ise en iyi ortalama ağırlıklandırılmış hatırlama performansı lnorm-tf parametreleri ile elde edilmiştir.



(a) (b)

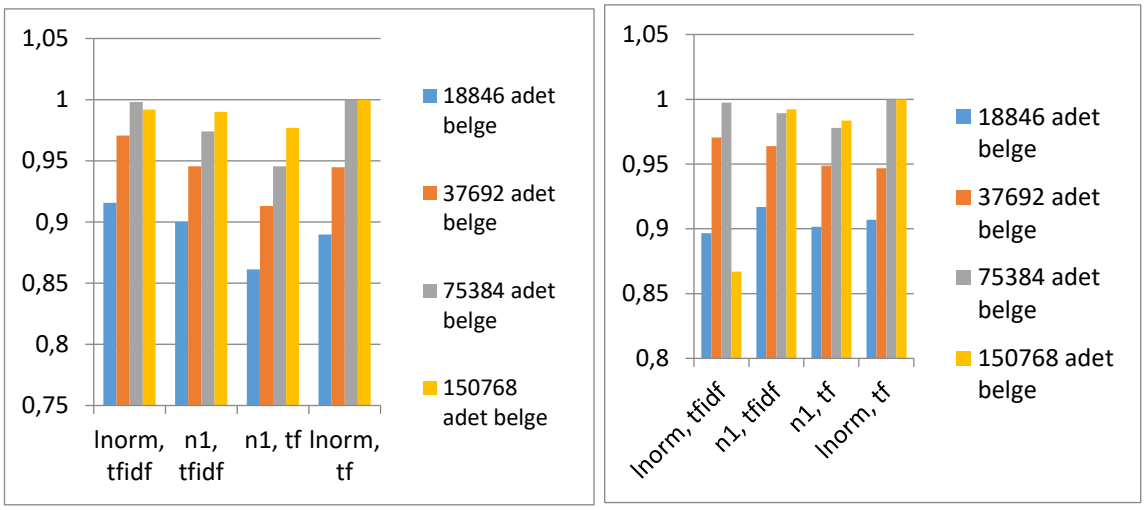
Şekil 4.8 Normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama ağırlıklandırılmış hatırlama değerleri
(a) Naif Bayes, (b) Tamamlayıcı Naif Bayes

Şekil 4.9'da Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan ortalama ağırlıklı hassasiyet değerleri karşılaştırımı verilmiştir. Bu şekle göre en iyi ortalama ağırlıklı hassasiyet değerinin 18846 adet dokümandan oluşan veri seti için Tamamlayıcı Naif Bayes algoritmasına; 37692 adet dokümandan oluşan veri seti için ise Naif Bayes algoritmasına ait olduğu görülmüştür. Algoritmaların 75384 ve 150768 adet dokümandan oluşan veri setleri için performansları eşittir.



Şekil 4.9 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan ortalama ağırlıklı hassasiyet değerleri karşılaştırımı

Şekil 4.10'da normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama ağırlıklı hassasiyet değerleri verilmiştir. **Şekil 4.10a**'ya göre en iyi ortalama ağırlıklı hassasiyet değeri 18846 ve 37692 adet dokümandan oluşan veri setleri için lnorm-tfidf parametreleriyle; 75384 ve 150768 adet dokümandan oluşan veri setleri için ise lnorm-tf parametreleriyle elde edilmiştir. **Şekil 4.10b**'ye göre en iyi ortalama ağırlıklı hassasiyet değeri 18846 adet dokümandan oluşan veri seti için n1-tfidf parametreleriyle; 37692 adet dokümandan oluşan veri seti için lnorm-tfidf parametreleriyle; 75384 ve 150768 adet dokümandan oluşan veri setleri için ise lnorm-tf parametreleriyle elde edilmiştir.



(a) (b)

Şekil 4.10 Normalizasyon ve ağırlık parametrelerine göre Naif Bayes ve Tamamlayıcı Naif Bayes algoritması en iyi ortalama ağırlıklı hassasiyet değerleri
(a) Naif Bayes, (b) Tamamlayıcı Naif Bayes

T test ve F test sonuçları **çizelge 4.11** ve **çizelge 4.12**'de verilmiştir. **Çizelge 4.11**'e göre t test sonuçları analiz edildiğinde ortalama doğruluk, ortalama ağırlıklandırılmış hatırlama ve ortalama ağırlıklandırılmış hassasiyet kriterleri için istatiksel değer kritik bir kuyruk değerinden küçük olduğundan ve p değeri 0.05'ten büyük olduğundan Naif Bayes ve Tamamlayıcı Naif Bayes Algoritmaları performansı birbirine benzerdir ve aralarında istatiktiksel olarak anlamlı fark yoktur. **Çizelge 4.12**'ye göre f test sonuçları analiz edildiğinde ortalama doğruluk, ortalama ağırlıklandırılmış hatırlama ve ortalama ağırlıklandırılmış hassasiyet kriterleri için istatiktiksel değer kritik bir kuyruk değerinden küçük olduğundan ve p değeri 0.05'ten büyük olduğundan Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları performansları birbirine benzerdir ve aralarında istatiktiksel olarak anlamlı fark yoktur.

Çizelge 4.11 T test sonuçları

	P değeri	Kritik bir kuyruk değeri	İstatiktiksel değer
Ortalama doğruluk	0.45575314	1.75305036	0.113029162
Ortalama ağırlıklı hatırlama	0.44361690	1.75305036	0.144236431
Ortalama ağırlıklı hassasiyet	0.391029643	1.75305036	-0.28165141

Çizelge 4.12 F test sonuçları

	P değeri	Kritik bir kuyruk değeri	İstatiktiksel değer
Ortalama doğruluk	0.24517807	0.69549154	0.07499835
Ortalama ağırlıklı hatırlama	0.25128596	0.70262898	0.09577654
Ortalama ağırlıklı hassasiyet	0.47219	1.03731178	-0.1710769

4.1.4 Değişen yumuşatma, normalizasyon, ağırlık parametreleri ve veri büyüklüğüne göre, Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan en iyi ortalama doğru sınıflanan örnek yüzdesi, ortalama eğitim ve ortalama test süreleri

Değişen yumuşatma, normalizasyon, ağırlık parametreleri ve veri büyüklüğüne göre her iki algoritmaya ait olan en iyi ortalama doğru sınıflanan örnek yüzdesi, ortalama eğitim ve ortalama test süreleri performans analizi **çizelge 4.13** ve **çizelge 4.14**'te verilmiştir. **Çizelge 4.13**'e göre en iyi ortalama eğitim süresi 18846 adet doküman içeren veri seti için n1-tf parametreleriyle; 37692 adet doküman içeren veriseti için lnorm-tf; 75384 adet doküman içeren veriseti için n1-tf; 150768 adet doküman içeren veriseti için n1-tf parametreleriyle elde edilmiştir. Genel olarak en iyi ortalama eğitim süreleri n1-tf parametreleriyle elde edilmiş olunup, veri setinin büyüklüğünün artırılması ortalama eğitim süresininde fazla bir değişiklik yapmamış, önce artıp sonra azalma eğilimi göstermiştir. En iyi ortalama test süresi 18846 adet doküman içeren veriseti için lnorm-tf parametreleriyle; 37692 adet doküman içeren veriseti için n1-tfidf parametreleriyle; 75384 adet doküman içeren veriseti için lnorm-tf; 150768 adet doküman içeren veriseti için lnorm-tf parametreleriyle elde edilmiştir. Genel olarak en iyi test süreleri lnorm-tf parametreleriyle elde edilmiş olunup,

veri setinin büyüklüğü arttıkça ortalama test süresi genel olarak önce artmış, sonra azalmıştır. En iyi doğru sınıflanan örnek yüzdesi 18846 adet doküman içeren veri seti için lnorm-tfidf parametreleriyle; 37692 adet doküman içeren veri seti için lnorm-tfidf parametreleriyle; 75384 adet doküman içeren veriseti için lnorm-tfidf parametreleriyle; 150768 adet doküman içeren veriseti için lnorm-tf parametreleriyle elde edilmiştir. Genel olarak, verisetinin büyüklüğünün artırılması doğru sınıflanan örnek yüzdesi performansını olumlu yönde etkilemiştir. **Çizelge 4.14**'e göre en iyi ortalama eğitim süresi 18846 adet dokümandan oluşan veri seti için n1-tf parametreleriyle; 37692 adet doküman içeren veri seti için n1-tf parametreleriyle; 75384 adet dokuman içerem veri seti için n1-tf parametreleriyle; 150768 adet doküman içeren veri seti için n1-tf parametreleriyle elde edilmiştir. En iyi ortalama test süresi 18846 adet dokümandan oluşan veri seti için lnorm-tf parametreleriyle; 37692 adet doküman içeren veri seti için lnorm-tfidf parametreleriyle; 75384 adet doküman içerem veri seti için lnorm-tf parametreleriyle; 150768 adet doküman içeren veri seti için lnorm-tf parametreleriyle elde edilmiştir. En iyi ortalama doğru sınıflanan örnek yüzdesi 18846 adet dokümandan oluşan veri seti için n1-tfidf parametreleriyle; 37692 adet doküman içeren veri seti için lnorm-tfidf parametreleriyle; 75384 adet dokuman içerem veri seti için lnorm-tf parametreleriyle; 150768 adet doküman içeren veri seti için lnorm-tf parametreleriyle oluşmuştur. Veri seti büyüklüklüğünün artması ortalama eğitim ve test sürelerini fazla artırmamıştır. Veri seti büyüklüğünün artırılması genel olarak ortalama doğru sınıflanan örnek yüzdesini pozitif yönde etkilemiştir.

Çizelge 4.13 Her bir veri seti ve her bir normalizasyon ve ağırlık parametresi için Naif Bayes algoritması ile elde edilen en iyi ortalama eğitim süresi, ortalama test süresi ve ortalama doğru sınıflanan örnek yüzdesi

Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
Lnorm normalizasyon, tfidf ağırlık	18846	8681.4	4722.6	91.14689
	37692	11572.6	6665.4	96.976
	75384	11134.8	5148.4	99.82282
	150768	11047	4518.4	99.1639
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
n1 normalizasyon, tfidf ağırlık	18846	7770.6	4841.5	89.00329
	37692	11130	6623	94.1226

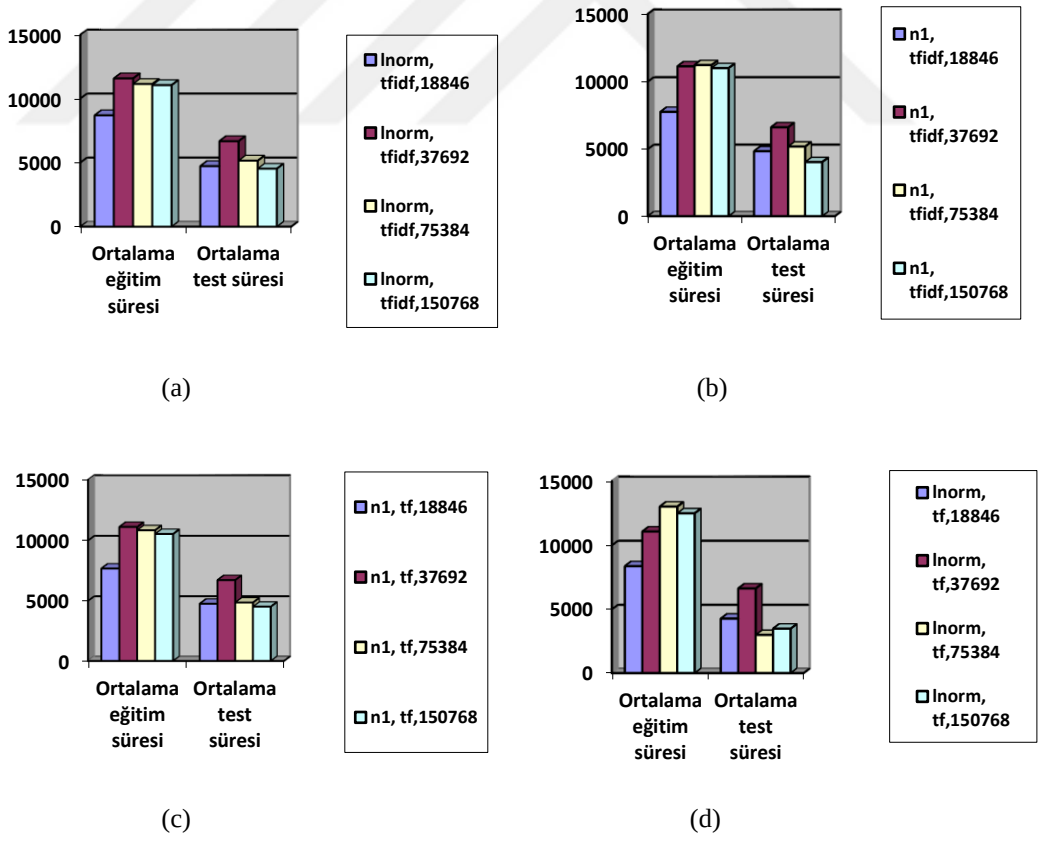
Çizelge 4.13 Her bir veri seti ve her bir normalizasyon ve ağırlık parametresi için Naif Bayes algoritması ile elde edilen en iyi ortalama eğitim süresi, ortalama test süresi ve ortalama doğru sınıflanan örnek yüzdesi (devam)

	75384	11221.5	5188.3	97.26648
	150768	10999.7	4032.2	98.98704
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
n1 normalizasyon, tf ağırlık	18846	7681.4	4755.6	83.97805
	37692	11111.3	6720.8	90.56828
	75384	10840.4	4857.9	93.94926
	150768	10529.8	4512.2	97.69051
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
lnorm normalizasyon, tf ağırlık	18846	8372.8	4270.6	86.78322
	37692	11073.4	6634.1	93.64076
	75384	13015	2989.5	99.79185

Çizelge 4.14 Her bir veri seti ve her bir normalizasyon ve ağırlık parametresi için Tamamlayıcı Naif Bayes algoritması ile elde edilen en iyi ortalama eğitim süresi, ortalama test süresi ve ortalama doğru sınıflanan örnek yüzdesi

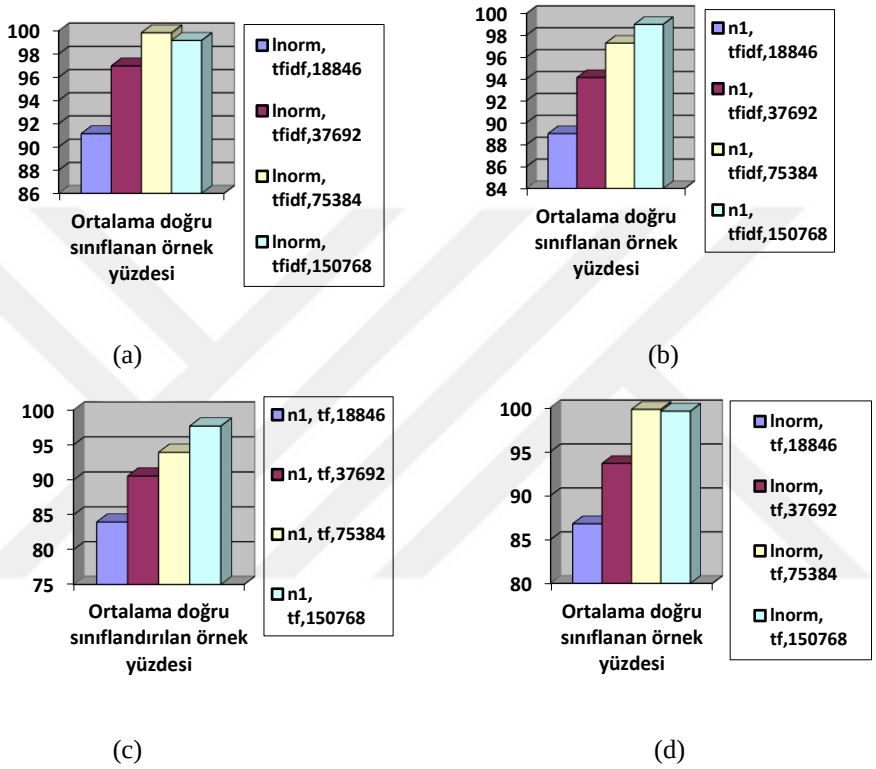
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
Lnorm normalizasyon, tfidf ağırlık	18846	10008.7	5524.1	89.50852
	37692	12990.6	6872.5	96.91629
	75384	12348.78	5581.444	99.75162
	150768	12358.8	4612.4	81.9466
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
n1 normalizasyon, tfidf ağırlık	18846	9294.9	5569.4	91.49856
	37692	12610.9	7616.1	96.33747
	75384	12338.1	5545.4	98.91801
	150768	12335.4	4609.6	99.22381
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
n1 normalizasyon, tf ağırlık	18846	9021.4	5498.7	89.79198
	37692	12383.6	7479.2	94.77751
	75384	12140.1	5442	97.72794
	150768	12018.1	4613.2	98.35264
Normalizasyon ve ağırlık parametreleri	**Belge sayısı**	**Ortalama eğitim süresi (ms)**	**Ortalama test süresi (ms)**	**Ortalama doğru sınıflanan örnek yüzdesi**
lnorm normalizasyon, tf ağırlık	18846	10468.4	3517.8	81.38633
	37692	12424	7544.3	94.34443
	75384	14239.2	2956.9	99.89593
	150768	14070.6	3539.8	99.59278

Şekil 4.11'de Naif Bayes algoritmasına ait olan normalizasyon, ağırlık parametresi ve veri seti büyüklüğüne göre en iyi ortalama eğitim ve test süreleri verilmiştir. **Şekil 4.11a**'ya göre en yüksek ortalama eğitim ve test süresi 37692 adet dokümandan oluşan veri seti için gerekli olmuştur. **Şekil 4.11b**'ye göre 75384 adet veriden oluşan verisetinin eğitimi için daha fazla süreye ihtiyaç duyulurken, 37692 adet dokümandan oluşan veri seti için en fazla test süresine ihtiyaç duyulmaktadır. **Şekil 4.11c**'de ise 37692 olan veri setini eğitmek ve test etmek için en fazla süreye gerek olduğu ve 37692 adet dokümandan sonra eğitim ve test sürelerinin azaldığı görülmektedir. **Şekil 4.11d**'de verildiğine göre en fazla ortalama eğitim süresi 75384 adet dokümanla olurken, en yüksek test süresi 37692 adet dokümanla olmuştur. Sonuç olarak **Şekil 4.11** için eğitim ve test sürelerinin veri setlerinin artışıyla lineer bir artış göstermediği, artışların çok düşük bir oranda gerçekleştiği, genel olarak veri seti büyüklüğünün artışıyla eğitim ve set süreleri önce artma sonra azalma eğilimi gösterdiği söylenebilmektedir.



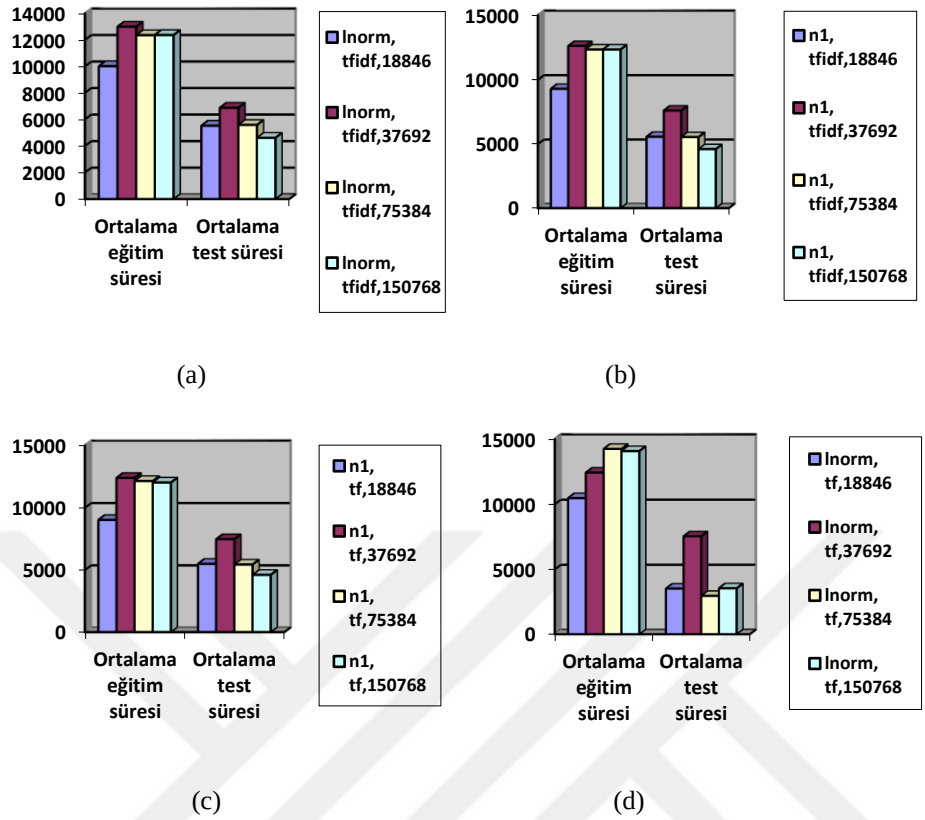
Şekil 4.11 Naif Bayes algoritmasına ait olan normalizasyon, ağırlık parametresi ve veri seti büyüklüğüne göre en iyi ortalama eğitim ve test süreleri (a) lnorm, tfidf (b) n1, tfidf (c) n1, tf (d) lnorm, tf

Şekil 4.12'de Naif Bayes algoritmasına ait olan normalizasyon, ağırlık parametresi ve veri seti büyüklüğüne göre en iyi ortalama doğru sınıflanan örnek yüzdesi verilmiştir. **Şekil 4.12a** ve **Şekil 4.12d**'de görüldüğü üzere 75384 adet dokümana sahip veri seti en fazla doğru sınıflanan örnek yüzdesine sahip olurken, **Şekil 4.12b** ve **Şekil 4.12c**'de veri seti büyüklüğünün artması ortalama doğru sınıflanan örnek yüzdesini artırmıştır.



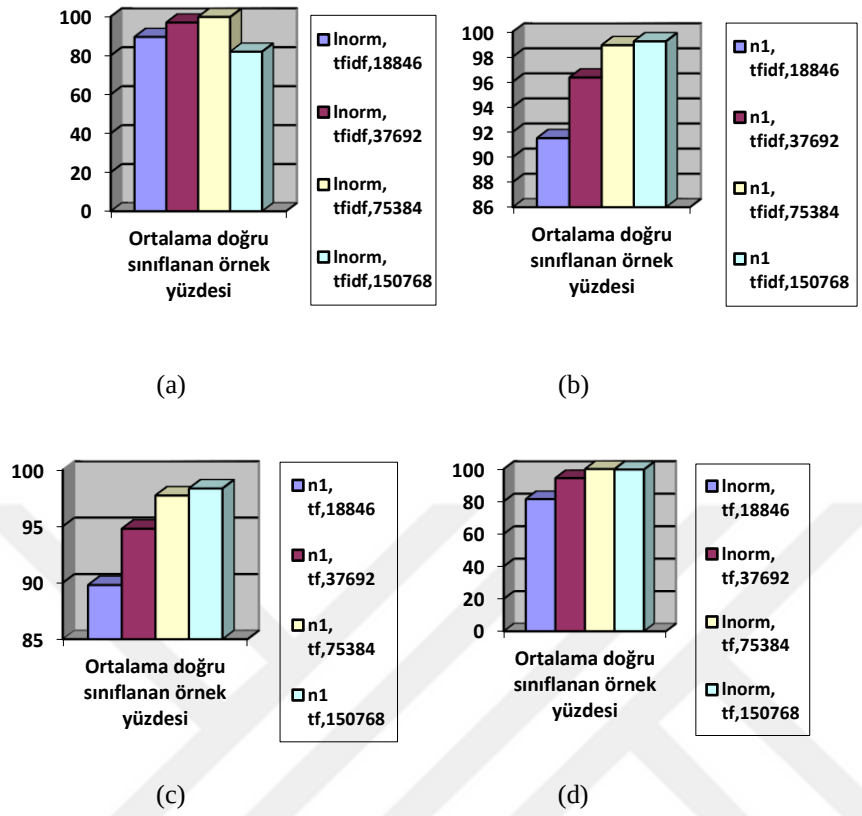
Şekil 4.12 Naif Bayes Algoritmasına ait aynı normalizasyon ve ağırlık parametresi ve farklı büyüklükteki veri setleri kullanıldığında en iyi ortalama doğru sınıflanan örnek yüzdesi
(a) lnorm, tfidf (b) n1, tfidf (c) n1, tf (d) lnorm, tf

Şekil 4.13'te Tamamlayıcı Naif Bayes algoritmasına ait olan normalizasyon, ağırlık parametresi ve veri seti büyüklüğüne göre en iyi ortalama eğitim ve test süreleri verilmiştir. Bu şekilde de görüldüğü üzere veri seti büyüklüğünün artırılması, ortalama eğitim ve test süresini fazla artırmamış olup, değerler birbirine çok yakındır. **Şekil 4.13b** ve **Şekil 4.13c**'de veri seti büyüklüğünün artırılması eğitim ve test sürelerinde önce artışa sonra da azalışa neden olmuştur. **Şekil 4.13d**'de ise normalizasyon parametresi olarak lnorm ve ağırlık parametresi olarak da tf 'nin kullanılması ortalama eğitim süresini diğer parametrelere göre daha fazla artırmıştır.



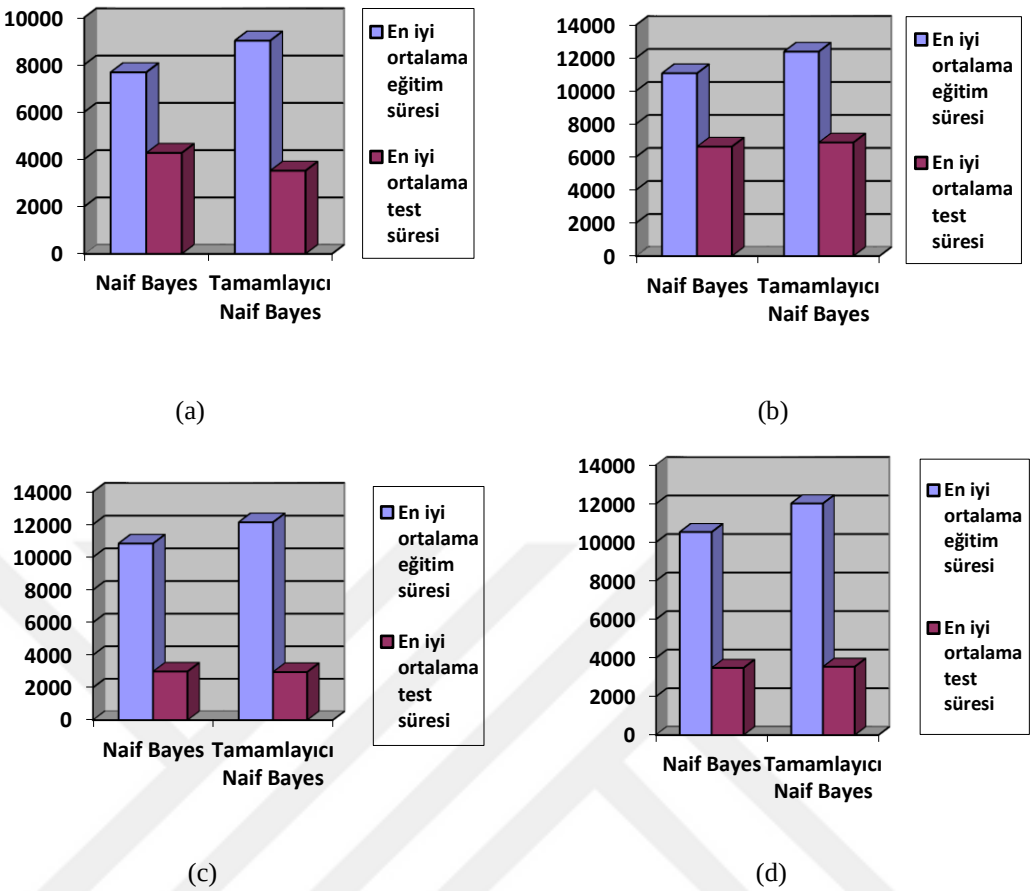
Şekil 4.13 Tamamlayıcı Naif Bayes Algoritmasına ait aynı normalizasyon ve ağırlık parametresi ve farklı büyüklükteki veri setleri kullanıldığında en iyi ortalama eğitim ve test süreleri
(a) lnorm, tfidf (b) n1, tfidf (c) n1, tf (d) lnorm, tf

Şekil 4.14'te Tamamlayıcı Naif Bayes algoritmasına ait olan normalizasyon, ağırlık parametresi ve veri seti büyüklüğüne göre en iyi ortalama doğru sınıflanan örnek yüzdesi gösterilmektedir. **Şekil 4.14a** ve **Şekil 4.14d**'de en iyi ortalama doğru sınıflanan örnek yüzdesi doküman sayısı 75384 olan veri setine aitken, **Şekil 4.14b** ve **Şekil 4.14c**'de doküman sayısı 150768 olan veri setine aittir.



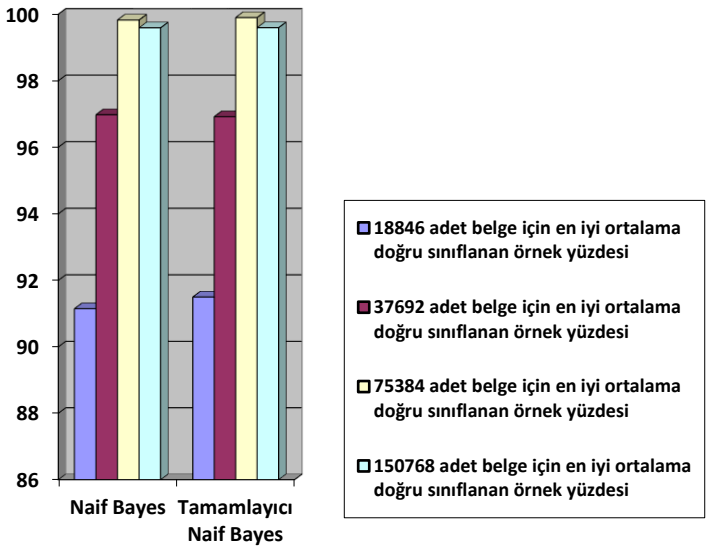
Şekil 4.14 Tamamlayıcı Naif Bayes Algoritmasına ait aynı normalizasyon ve ağırlık parametresi ve farklı büyüklükteki veri setleri kullanıldığında en iyi ortalama doğru sınıflanan örnek yüzdesi
(a) lnorm, tfidf (b) n1, tfidf (c) n1, tf (d) lnorm, tf

Şekil 4.15'te farklı büyüklükteki veri setleri için Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama eğitim ve test sürelerinin karşılaştırılması verilmiştir. Bu şekle göre tüm veri setleri için ortalama eğitim süresi Tamamlayıcı Naif Bayes algoritması için en fazladır. 18846 ve 75384 adet dokümandan oluşan veri setleri için Tamamlayıcı Naif Bayes test süresi daha azken, 37692 ve 150768 adet dokümandan oluşan veri setleri için Naif Bayes test süresi daha azdır.



Şekil 4.15 Farklı büyüklükteki veri setleri için Naif Bayes ve Tamamlayıcı Naif Bayes en iyi ortalama eğitim ve test sürelerinin karşılaştırılması
(a) 18846 adet doküman, (b) 37692 adet doküman, (c) 75384 adet doküman, (d) 150768 adet doküman

Şekil 4.16'da tüm veri setleri için Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan ortalama en doğru sınıflanan örnek yüzdesi verilmiştir. Bu şekle göre doküman sayısı 37692 olan veri seti dışında diğer tüm veri setleri için en iyi ortalama doğru sınıflanan örnek yüzdesi Tamamlayıcı Naif Bayes'e aittir.



Şekil 4.16 Tüm veri setleri için Naif Bayes ve Tamamlayıcı Naif Bayes'e ait olan ortalama en doğru sınıflanan örnek yüzdesi

4.2 Duygu Analizi

Bu bölümde Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarına ait olan duygu analizi performansları sonuçlarına yer verilmiş, yapılan analizin doğruluk derecesi gösterilmiştir.

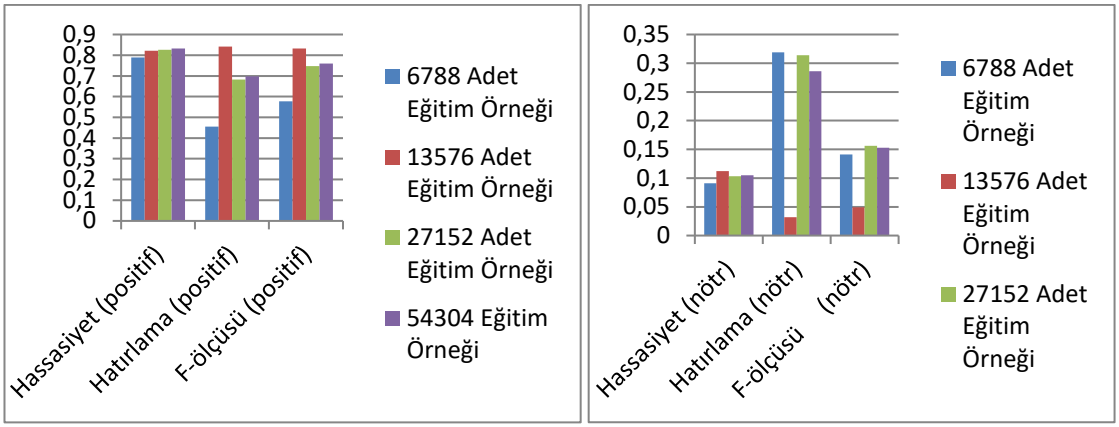
4.2.1 Naif Bayes sonuçları

Çizelge 4.15'te her bir eğitim veri setinin oluşturduğu hassasiyet, hatırlama ve F-ölçüsü değerleri verilmiştir. Bu çizelgeye göre, pozitif sınıfı için en iyi hassasiyet değeri en büyük veri seti olan 54304 adet örnekten oluşan eğitim setiyle; en iyi hatırlama değeri 13576 adet eğitim örneği içeren veri setiyle; en iyi F ölçüsü değeri 13576 adet örnek içeren veri setiyle oluşmuştur. Nötr sınıfı için en iyi hassasiyet değeri 13576 adet örnekten oluşan eğitim setiyle; en iyi hatırlama değeri 6788 adet eğitim örneği içeren veri setiyle; en iyi F ölçüsü değeri 27152 adet örnek içeren veri setiyle elde edilmiştir. Negatif sınıfı için en iyi hassasiyet değeri 27152 adet örnekten oluşan eğitim setiyle; en iyi hatırlama değeri 13576 adet eğitim örneği içeren veri setiyle; en iyi F ölçüsü değeri 13576 adet örnek içeren veri setiyle oluşmuştur. En iyi genel doğruluk değeri eğitim seti olarak 13576 adet örnekten oluşan veri seti kullanıldığında gözlenmiştir.

Çizelge 4.15 Tüm veri setleri için Naif Bayes hassasiyet, hatırlama, F-ölçüsü, genel doğruluk sonuçları

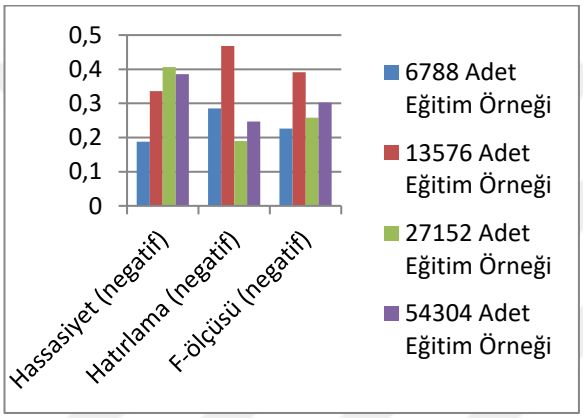
		Pozitif	Nötr	Negatif
6788 adet eğitim örneği	**Hassasiyet**	0.789	0.091	0.188
	Hatırlama	0.456	0.319	0.285
	F ölçüsü	0.578	0.141	0.226
	Genel doğruluk	0.419		
		Pozitif	**Nötr**	**Negatif**
13576 adet eğitim örneği	**Hassasiyet**	0.822	0.112	0.336
	Hatırlama	0.841	0.032	0.468
	F ölçüsü	0.832	0.049	0.391
	Genel doğruluk	0.709		
		Pozitif	**Nötr**	**Negatif**
27152 adet eğitim örneği	**Hassasiyet**	0.827	0.103	0.406
	Hatırlama	0.683	0.314	0.190
	F ölçüsü	0.748	0.156	0.258
	Genel doğruluk	0.578		
		Pozitif	**Nötr**	**Negatif**
54304 adet eğitim örneği	**Hassasiyet**	0.832	0.105	0.386
	Hatırlama	0.698	0.286	0.247
	F ölçüsü	0.759	0.153	0.302
	Genel doğruluk	0.60		

Şekil 4.17'de her bir eğitim veri setiyle oluşan performanslar grafiksel olarak verilmiştir. **Şekil 4.17a**'da görüldüğü üzere eğitim seti arttıkça hassasiyet değeri artmıştır. Yani yanlış olarak pozitif sınıfına konan örnek sayısı azalmıştır. En iyi hassasiyet değeri en fazla örneğe sahip olan veri setiyle elde edilirken, en iyi hatırlama ve F-ölçüsü değeri 13576 adet veri içeren veri setiyle oluşmuştur. **Şekil 4.17b**'de performans diğer sınıflamaya göre daha düşüktür. Bu şekle göre en iyi hassasiyet değeri 13576 adet doküman içeren veri setiyle; en iyi hatırlama değeri 6788 adet eğitim örneği içeren veri setiyle; en iyi F-ölçüsü değeri 27152 adet eğitim örneği içeren veri seti kullanıldığında elde edilmiştir. **Şekil 4.17c**'de hassasiyet değeri veri seti büyüklüğü ile artış göstermiş diyebiliriz. Bu artış pozitif sınıfı için daha düzenliydi. Yani en yüksek hassasiyet değeri burada 27352 adet örnekle sağlanırken, pozitif sınıfında 54304 adet örnekle sağlanmıştı. En yüksek hatırlama değeri pozitif sınıfındaki gibi 13576 adet örnekle sağlanmıştır. F ölçüsü dağılımı ise pozitif sınıfının benzeridir, en iyi F ölçüsü yine büyüklüğü 13576 adet olan eğitim setiyle elde edilmiştir.



(a)

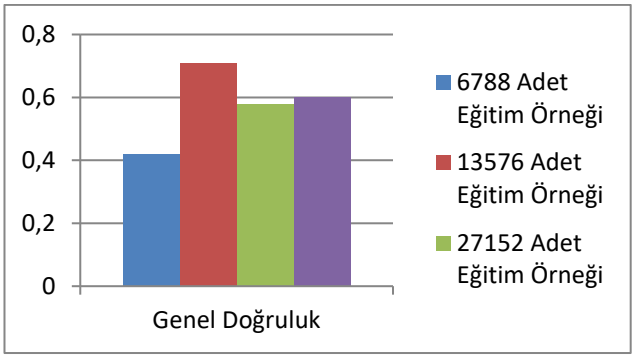
(b)



(c)

Şekil 4.17 Her bir eğitim veri setiyle oluşan performanslar
(a) Pozitif sınıfı, (b) Nötr sınıfı, (c) Negatif sınıfı

Şekil 4.18'de her bir eğitim seti için genel doğruluk değerleri verilmiştir. Bu şekle göre en iyi genel doğruluk performansı 13576 adet eğitim örneği içeren veri seti kullanıldığında elde edilmiştir ve eğitim seti büyüklüğündeki artış genel anlamda genel doğruluk değerini olumlu etkilemiştir.



Şekil 4.18 Her bir eğitim seti için genel doğruluk değerleri

4.2.2 Tamamlayıcı Naif Bayes sonuçları

Bu bölümde Tamamlayıcı Naif Bayes algoritmasına ait olan sonuçlar çizelge ve şekiller yardımıyla gösterilmiştir. **Çizelge 4.16**'da tüm veri setleri için Tamamlayıcı Naif Bayes algoritmasına ait olan hassasiyet, hatırlama, F-ölçüsü ve tüm doğruluk değerleri sonuçları verilmiştir. Bu çizelgeye göre pozitif sınıfı için en iyi hassasiyet değeri 27152 adet eğitim örneği içeren veri setiyle; en iyi hatırlama ve f-ölçüsü değerleri 13576 adet örnek içeren veri setiyle elde edilmiştir. Nötr sınıfı için en iyi hassasiyet değeri 13576 adet eğitim örneği içeren veri setinde; en iyi hatırlama ve f-ölçüsü değerleri 27152 adet örnek içeren veri setinde görülmüştür. Negatif sınıfı için en iyi hassasiyet değeri 27152 adet eğitim örneği içeren veri setiyle; en iyi hatırlama değeri 6788 adet örnek içeren veri setiyle; en iyi f-ölçüsü ise 13576 adet örnek içeren veri setiyle elde edilmiştir.

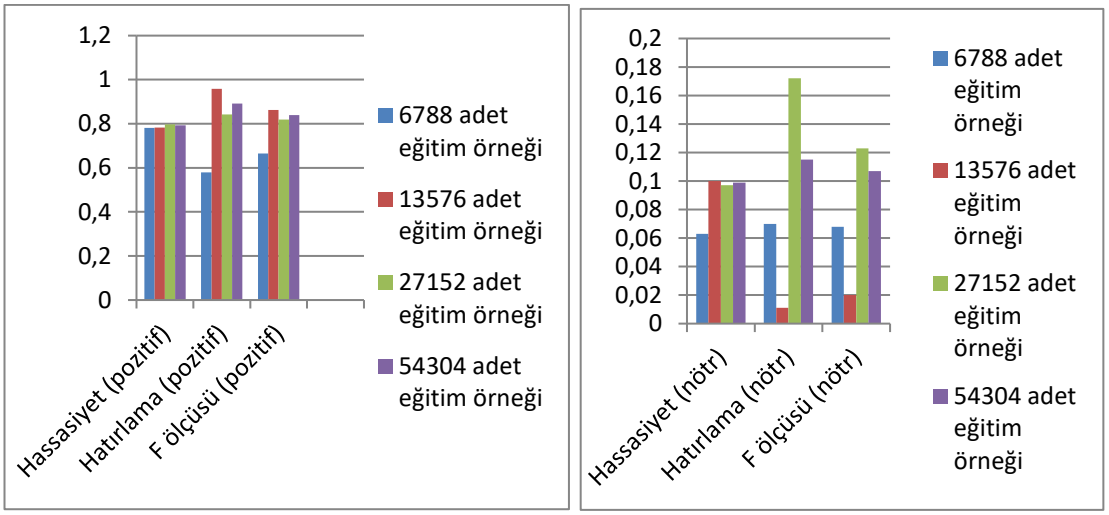
Çizelge 4.16 Tüm veri setleri için Tamamlayıcı Naif Bayes algoritmasına ait olan hassasiyet, hatırlama, F-ölçüsü, tüm doğruluk değerleri sonuçları

		Pozitif	**Nötr**	**Negatif**
6788 adet eğitim örneği	**Hassasiyet**	0.781	0.063	0.178
	Hatırlama	0.579	0.070	0.421
	F ölçüsü	0.665	0.068	0.250
	Tüm doğruluk	0.506		
		Pozitif	**Nötr**	**Negatif**
13576 adet eğitim örneği	**Hassasiyet**	0.783	0.100	0.438
	Hatırlama	0.958	0.011	0.180
	F ölçüsü	0.862	0.02	0.256
	Tüm doğruluk	0.756		

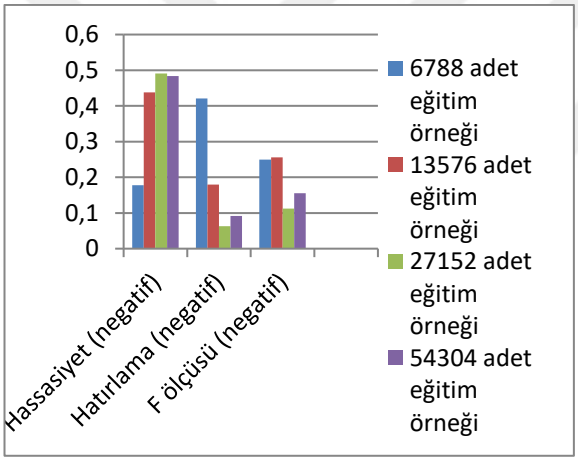
Çizelge 4.16 Tüm veri setleri için Tamamlayıcı Naif Bayes algoritmasına ait olan hassasiyet, hatırlama, F-ölçüsü, tüm doğruluk değerleri sonuçları (devam)

		Pozitif	Nötr	Negatif
27152 adet eğitim örneği	**Hassasiyet**	0.797	0.097	0.491
	Hatırlama	0.842	0.172	0.063
	F ölçüsü	0.819	0.123	0.112
	Tüm doğruluk	0.668		
		Pozitif	**Nötr**	**Negatif**
54304 adet eğitim örneği	**Hassasiyet**	0.793	0.099	0.484
	Hatırlama	0.891	0.115	0.092
	F ölçüsü	0.839	0.107	0.155
	Tüm doğruluk	0.704		

Şekil 4.19'da her bir sınıfa ait olan Tamamlayıcı Naif Bayes algoritması performansları verilmiştir. **Şekil 4.19a**'ya göre en iyi hassasiyet değeri 27152 adet eğitim setinden oluşan veri setiyle elde edilmiştir. Naif Bayes algoritmasında ise 54304 adet veriden oluşan veri setiyle elde edilmişti. En iyi hatırlama ve F ölçüsü değerleri 13576 adet veriden oluşan veri setiyle elde edilmiştir. Pozitif sınıfı için Naif Bayes ve Tamamlayıcı Naif Bayes performans eğilimleri benzerdir. **Şekil 4.19b**'ye göre en iyi hassasiyet değeri 13576 adet örnek içeren veri setine ve en yüksek sayıda eğitim setine sahip olan veri seti ile elde edilmiştir. En iyi hatırlama değeri 27152 adet veriden oluşan veri setiyle elde edilmiştir. En iyi F ölçüsü değerine Naif Bayes algoritmasındaki gibi 27152 adet veriden oluşan veri seti ile ulaşılmıştır. **Şekil 4.19c**'de negatif sınıfına ait olan performanslara yer verilmiştir. Bu şekle göre veri setini 27152 adet verinin üstüne çıkarmak hassasiyet değerini olumsuz yönde etkilemiştir. Naif Bayes negatif sınıflandırma ve Tamamlayıcı Naif Bayes pozitif sınıflandırmada da durum aynıdır. En iyi hatırlama değeri en küçük büyüklükte olan veri setine aittir. En iyi F ölçüsü değeri 13576 adet veriden oluşan veri setine aittir. Naif Bayes pozitif ve negatif sınıflandırmada ve Tamamlayıcı Naif Bayes pozitif sınıflandırmada da durum aynıdır.



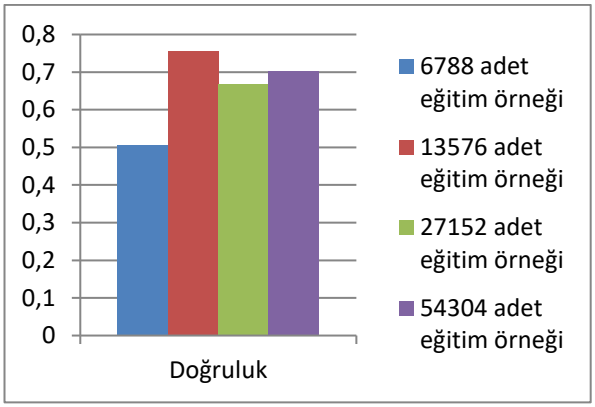
(a) (b)



(c)

Şekil 4.19 Her bir sınıfa ait olan Tamamlayıcı Naif Bayes algoritması performansları
(a) Pozitif Sınıfı, (b) Nötr Sınıfı, (c) Negatif Sınıfı

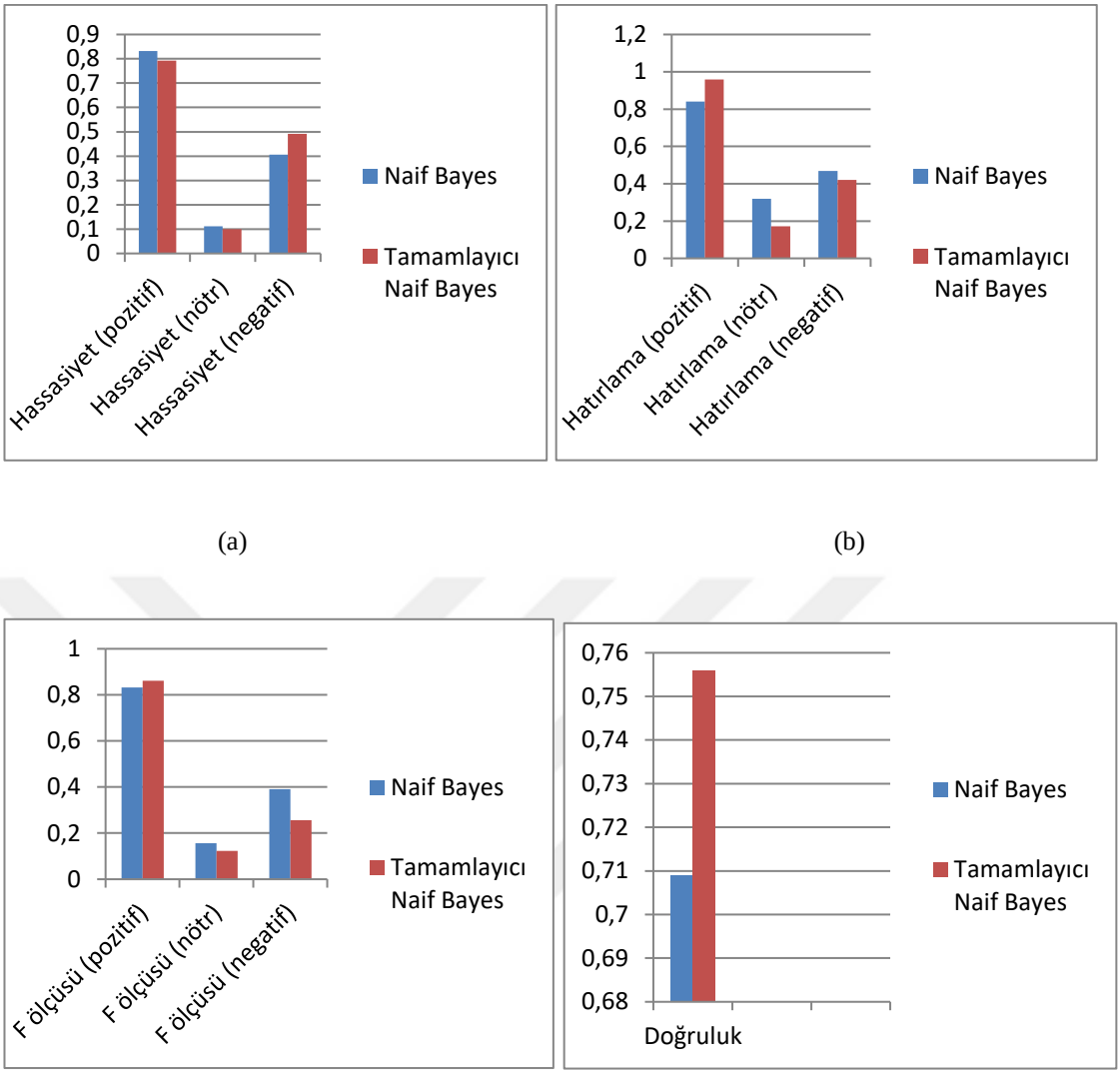
Şekil 4.20'de Tamamlayıcı Naif Bayes sınıflama algoritması ile oluşan genel doğruluk değerleri karşılaştırılmıştır. Bu şekle göre en iyi genel doğruluk değeri 13576 adet eğitim örneğinden oluşan veri setinde oluşmuştur ve genel doğruluk değerleri eğilimleri Naif Bayes algoritmasının sonuçları ile benzerdir.



Şekil 4.20 Tamamlayıcı Naif Bayes sınıflama algoritması ile oluşan genel doğruluk değerleri

4.2.3 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile elde edilen en iyi performansların karşılaştırımı

Bu bölümde Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları ile elde edilen en iyi performansların karşılaştırımına yer verilmiştir. **Şekil 4.21**'de bu iki algoritmaya ait olan hassasiyet, hatırlama, F-ölçüsü ve doğruluk değerleri karşılaştırılmıştır. **Şekil 4.21a**'ya göre pozitif ve nötr kelimelerine ait olan en iyi hassasiyet değeri Naif Bayes algoritması ile elde edilirken, negatif sınıfına ait olan en iyi hassasiyet değerini Tamamlayıcı Naif Bayes algoritması vermiştir. **Şekil 4.21b**'de de görüldüğü üzere nötr ve negatif sınıfına ait en iyi hatırlama değeri Naif Bayes algoritması ile oluşurken, pozitif hatırlama en iyi değeri Tamamlayıcı Naif Bayes algoritması ile oluşmuştur. **Şekil 4.21c**'de ise F-ölçüsü performansları karşılaştırılmış olup, pozitif dışındaki tüm sınıflar için en iyi F ölçüsü değerini Naif Bayes algoritması göstermiştir. **Şekil 4.21d**'de doğruluk performans değerlendirme kriterine göre karşılatırma yapılmış, Tamamlayıcı Naif Bayes algoritmasının performansının daha iyi olduğu gözlenmiştir. Sonuç olarak, hassasiyet (pozitif), hassasiyet (nötr), hatırlama (nötr), hatırlama (negatif), F ölçüsü (nötr), F ölçüsü (negatif) en iyi değerleri Naif Bayes algoritmasına aitken; hassasiyet (negatif), hatırlama (pozitif), F ölçüsü (pozitif) ve genel doğruluk değerleri en iyi Tamamlayıcı Naif Bayes algoritmasına aittir.



Şekil 4.21 Naif Bayes ve Tamamlayıcı Naif Bayes algoritmaları performans karşılaştırımı
(a) Hassasiyet, (b) Hatırlama, (c) F ölçüsü, (d) Doğruluk

4.2.4 "Amozon movie reviews" veri setindeki duygular ve verilen oylar arasındaki ilişki

Duygular ve verilen oylar arasında bir ilişkinin olduğu lojistik regresyon sınıflama algoritması ile görülmüş, pozitif, negatif ve nötr duyguları ile verilen oylar arasındaki ilişki lojistik regresyon sınıflama algoritması ile tespit edilmiştir. Lojistik regresyon, bir olayın olma olasılığını tahmin etmeye ve bir bağımlı verinin bir ya da birden çok bağımsız veriyle

ilişkisini açıklamaya yaramaktadır. Örneğin; bir kişinin bir ürünü alıp almayacağı ya da piyasaya yeni sürülen bir ürünün beğenilip beğenilmeyeceği gibi durumlar lojistik regresyon ile tahmin edilebilmektedir.

Bir önceki bölümde en iyi doğruluk değerinin 13576 adet eğitim örneğinden oluşan veri setinin kullanılmasıyla oluştuğu görülmüştü. Bu bölümde en iyi genel doğruluk değerini veren 13576 adet eğitim örneğinin kullanılmasıyla sınıflanan "Amazon movie reviews" veri seti kullanılmıştır.

Bu çalışmada, lojistik regresyon algoritması ikili sınıflaması seçilmiş, veri setindeki nötr sınıfı pozitif veya negatif sınıfına çevrilmiştir. 1 ve 2 olarak oylanan nötr sınıfına ait olan veriler negatif sınıfına; 3, 4 ve 5 olarak oylanan ve nötr sınıfına konan veriler pozitif sınıfına alınmıştır. Verilen oylar ile duygular arasındaki ilişki gözlenmiştir.

Verinin rastgele %70'i eğitim setini oluştururken, geriye kalan %30'u test setini oluşturmuştur. Bu şekilde 3 farklı veri seti oluşturulmuştur. Herbir deney bu veri setleriyle 3 kez yapılmış ve sonuçların ortalaması alınmıştır.

Sınıflama 10 iterasyon sayısı boyunca, 0.0001 lambda (coefficient decay) ile 0.001, 0.0001 ve 0.00001 öğrenme katsayısı parametreleri boyunca denenmiştir. En iyi Eğri Altındaki Alan (Area Under Curve (AUC)) değeri 0.703 olarak bulunmuştur. Bu değerin 1'e yakın olması daha fazla bir ilişkiyi ifade etmektedir.

Çizelge 4.17 Lojistik regresyon sonuçları

Öğrenme katsayısı	**Ortalama eğri altındaki alan**	**Ortalama doğruluk**
0.001	0.7	0.9398
0.0001	0.703	0.9398
0.00001	0.7	0.9398

Çizelge 4.17'ye göre çok iyi bir doğruluk değeri elde edilmiştir. Ortalama eğri altındaki alan 0.5 değerinden büyüktür. En iyi performans, öğrenme katsayısı 0.0001 seçildiğinde

elde edilmiştir. Bu çizelgeye göre oylar ve duygular arasında çok yakın bir ilişki vardır yorumu yapılabilmektedir.

4.3 Öneri Sistemi Geliştirilmesi

Oluşturulan modellerin kullanılan veri setlerine ve benzerlik ölçütlerine göre değişen performansları **çizelge 4.18** ve **çizelge 4.19**'da verilmiştir. **Çizelge 4.18**'de verildiğine göre ortalama mutlak hata ve kök ortalama kare hatası performans değerlendirme kriterlerine göre en iyi performansı log benzerlik mesafe ölçüsü gösterirken; hassasiyet, hatırlama ve f-ölçütü performans değerlendirme kriterlerine göre en iyi performansı Pearson mesafe ölçüsü göstermiştir. **Çizelge 4.19**'a göre ortalama mutlak hata ve kök ortalama kare hatası baz alındığında en iyi performansın merkezlenmemiş kosinüs mesafe ölçüsüne ait olduğu görülmektedir. Hassasiyet, hatırlama ve f-ölçüsü benzerlik ölçütleri bazında ise en iyi performans Öklid mesafe ölçüsü kullanıldığında elde edilmiştir. **Çizelge 4.18** ve **çizelge 4.19** karşılaştırıldığında en iyi ortalama mutlak hata ve kök ortalama kare hatası değerlerinin merkezlenmemiş kosinüs mesafe ölçüsü ile elde edildiği görülmüştür. En iyi hassasiyet, hatırlama ve f-ölçüsü değerleri ise Pearson ile elde edilmiştir. Ayrıca, veri setinin büyüklüğü artırıldığında ortalama mutlak hata ve kök ortalama kare hatası değerlerinde iyileşme olduğu görülmüştür.

Çizelge 4.18 100,000 adet oydan oluşan veri seti için sonuçlar

	Pearson	**Öklid**	**Log Benzerlik**	**Merkezlenmemiş Kosinüs**
Ortalama Mutlak Hata	0.903667	0.894758	**0.815834**	0.830392
Kök Ortalama Karesel Hatası	1.122772	1.123979	**1.037744**	1.04070
Hassasiyet	**0.015982**	0.014580	0.000507	0.000253
Hatırlama	**0.0148900**	0.012035	0.000508	0.000254
F-ölçüsü	**0.015417**	0.013186	0.000508	0.000254

Çizelge 4.19 1 milyon adet oydan oluşan veri seti için sonuçlar

	Pearson	**Öklid**	**Log Benzerlik**	**Merkezlenmemiş Kosinüs**
Ortalama Mutlak Hata	0.812165	0.789041	0.738430	**0.728047**
Kök Ortalama Karesel Hatası	1.044703	1.03509	0.9579285	**0.951552**
Hassasiyet	0.002436	**0.005986**	0.000013	0.000018
Hatırlama	0.002442	**0.005589**	0.000010	0.000011
F-ölçüsü	0.00244	**0.005936**	0.000011	0.000010

4.4 Öneri Sistemlerinin Tahmin Performansını İyileştirmek İçin Evrimsel Sinir Ağları

Movielens 100K veri seti için genetik algoritma öncesi ve sonrası ortalama kare hata sonuçları **çizelge 4.20**, **çizelge 4.21** ve **çizelge 4.22**'de verilmiştir. **Çizelge 4.20**'ye göre çok katmanlı algılayıcı kullanıldığında momentum ve gradyan öğrenme kuralları için 3 ve 5 gizli katmanla optimizasyon sayesinde iyileşme olduğu görülmektedir. **Çizelge 4.21**'e göre genelleştirilmiş ileri beslemeli ağ kullanıldığında momentum öğrenme kuralı ile 5 gizli katman, gradyan öğrenme kuralı ile 3 ve 5 gizli katman kullanıldığında iyileşme olmuştur. **Çizelge 4.22**'ye göre koaktif nöro bulanık çıkarım sistemi kulanıldığında genetik algoritma ile bir iyileşme gözlenmemiştir.

Çizelge 4.20 Movielens 100K veri seti için çok katmanlı algılayıcı ile oluşan ortalama kare hata değerleri

	Çok Katmanlı Algılayıcı					
	Momentum			**Gradyan**		
	3 gizli katman	**5 gizli katman**	**7 gizli katman**	**3 gizli katman**	**5 gizli katman**	**7 gizli katman**
Optimizasyondan önce	0.31728	0.26993	0.25796	0.29472	0.47740	0.26429
Optimizasyon	**0.25788**	**0.25735**	0.25796	**0.24838**	**0.26872**	0.26429

Çizelge 4.21 Movielens 100K veri seti için genelleştirilmiş ileri besleme ile oluşan ortalama kare hata değerleri

	Genelleştirilmiş İleri Besleme					
	Momentum			**Gradyan**		
	3 gizli katman	**5 gizli katman**	**7 gizli katman**	**3 gizli katman**	**5 gizli katman**	**7 gizli katman**
Optimizasyondan önce	0.40421	0.61841	0.29510	1.15396	0.70673	0.79529
Optimizasyon	0.40421	**0.45736**	0.29510	**0.46800**	**0.58902**	0.79529

Çizelge 4.22 Movielens 100K veri seti için koaktif nöro bulanık çıkarım sistemi ile oluşan ortalama kare hata değerleri

	Koaktif Nöro Bulanık Çıkarım Sistemi			
	TSK		**Tsukamoto**	
	Momentum	**Gradyan**	**Momentum**	**Gradyan**
Optimizasyondan Önce	0.25219	0.25587	0.33036	0.28264
Optimizasyon	0.25219	0.25587	0.33036	0.28264

Movielens 1M veriseti için ortalama kare hata sonuçları **çizelge 4.23**, **çizelge 4.24** ve **çizelge 4.25**'te verilmiştir. **Çizelge 4.23**'e göre çok katmanlı algılayıcı kullanıldığında momentum kuralı ile 3 ve 5 gizli katman, gradyan öğrenme kutalı ile 3, 5 ve 7 gizli katman kullanıldığında genetik algoritma ile iyileşme görülmüştür. **Çizelge 4.24**'e göre ileri beslemeli ağ kullanıldığında momentum kuralı ile 5 ve 7 gizli katman, gradient kuralı ile 3, 5 ve 7 gizli katman kullanıldığında iyileşme olmuştur. **Çizelge 4.25**'ya göre Takagi-Sugeno-Kang bulanık modeli kullanılığında gradyan öğrenme kuralı ile, Tsukamoto bulanık modeli kullanıldığında momentum öğrenme kuralı ile iyileşme görülmüştür.

Çizelge 4.23 Movielens 1M veri seti için çok katmanlı algılayıcı ile oluşan ortalama kare hata değerleri

	Çok Katmanlı Algılayıcı					
	Momentum			**Gradyan**		
	3 gizli katman	**5 gizli katman**	**7 gizli katman**	**3 gizli katman**	**5 gizli katman**	**7 gizli katman**
Optimizasyondan önce	0.37336	0.25774	0.25824	0.27702	0.38233	0.27054
Optimization	**0.24820**	**0.25680**	0.25824	**0.24948**	**0.25092**	**0.24955**

Çizelge 4.24 Movielens 1M veri seti için genelleştirilmiş ileri besleme ile oluşan ortalama kare hata değerleri

	Genelleştirilmiş İleri Besleme					
	Momentum			**Gradyan**		
	3 gizli katman	**5 gizli katman**	**7 gizli katman**	**3 gizli katman**	**5 gizli katman**	**7 gizli katman**
Optimizasyondan önce	0.21458	0.26927	0.29676	0.32991	0.70883	0.84200
Optimizasyon	0.21458	**0.24986**	**0.25639**	**0.25490**	**0.31410**	**0.28433**

Çizelge 4.25 Movielens 1M veri seti için koaktif nöro bulanık çıkarım sistemi ile oluşan ortalama kare hata değerleri

	Koaktif Nöro Bulanık Çıkarım Sistemi			
	TSK		**Tsukamoto**	
	Momentum	**Gradyan**	**Momentum**	**Gradyan**
Optimizasyondan Önce	0.25825	1.09930	0.37259	0.28741
Optimizasyon	0.25825	**0.43654**	**0.33754**	0.28741

Deneyler sonucunda elde edilen en iyi ortalama kare hata ve kök ortalama karesel hata değerleri **çizelge 4.26** ve **çizelge 4.27**'de verilmiştir. **Çizelge 4.26**'ya göre Movielens 100K veri seti kullanıldığında en iyi ortalama kare hata değeri çok katmanlı algılayıcıya genetik algoritma uygulandığında (3 gizli katman, gradyan öğrenme kuralı) elde edilmiştir. **Çizelge 4.27**'de görüldüğü üzere Movielens 1M veri seti kullanıldığında en iyi ortalama kare hata değeri genelleştirilmiş ileri beslemeli ağ (3 gizli katman, momentum öğrenme kuralı) kullanıldığında oluşmuştur.

Çizelge 4.26 Movielens 100K veri seti kullanılarak elde edilen en iyi ortalama kare hata ve kök ortalama kare hata değerleri

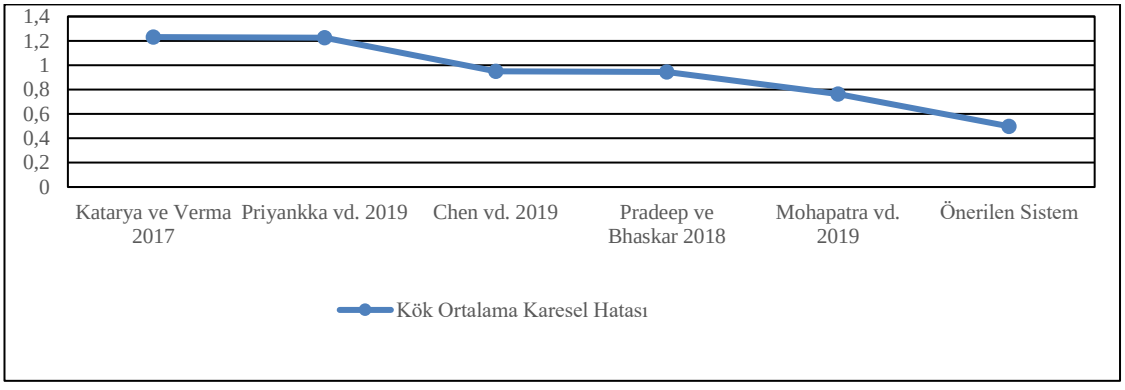
Movielens 100K veri seti		
Metot	**Ortalama Kare Hata**	**Kök Ortalam Kare Hata**
Çok Katmanlı Algılayıcı +Genetik algoritma (3 gizli katman, gradyan)	0.24838	0.49838

Çizelge 4.27 Movielens 1M veri seti kullanılarak elde edilen en iyi ortalam kare hata ve kök ortalama kare hata değerleri

1M veri seti		
Metot	**Ortalama Kare Hata**	**Kök Ortalama Kare Hata**
Genelleştirilmiş İleri Besleme (3 gizli katman, momentum)	0.21458	0.46322

4.4.1 Elde edilen deney sonuçlarının diğer çalışmalarla karşılaştırımı

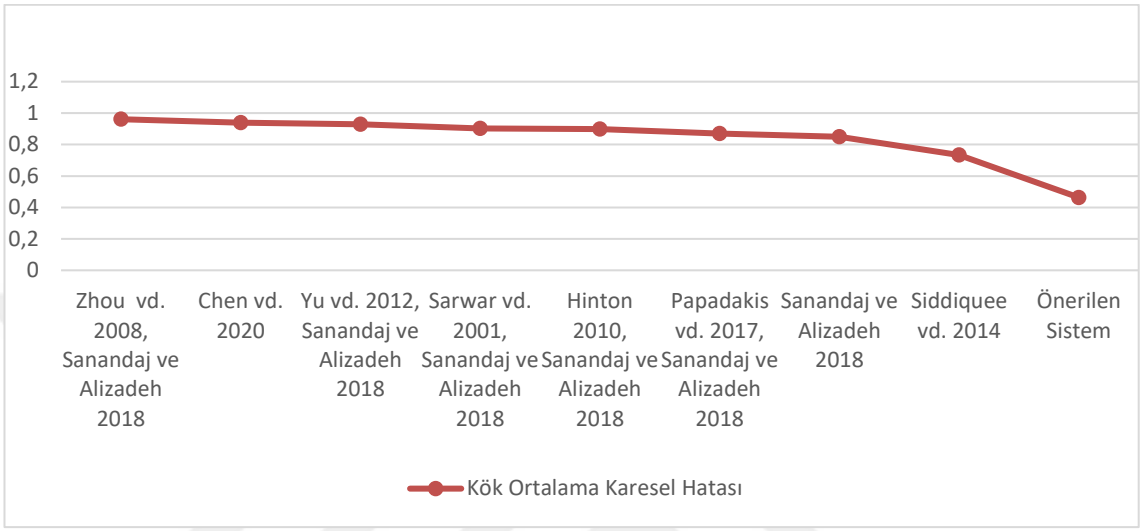
Pradeep ve Bhaskar (2018), Chen ve arkadaşları (2019), Katarya ve Verma (2017), Mohapatra ve arkadaşları (2019) ve Priyankka ve arkadaşları (2019) Movielens 100K veri setini kullanmış, önerdikleri öneri sisteminin performansını kök ortalama kare hatası değerlendirme kriteri ile ölçmüşlerdir. Pradeep ve Bhaskar SVD yaklaşımını, Chen ve arkadaşları UCEC&CF algoritmasını, Katarya ve Verma K-ortalama Cuckoo algoritmasını, Mohapatra ve arkadaşları ve k-means clustering ve cuckoo arama optimizasyon algoritmasını kullanan bir öneri sistemini, Priyankka ve arkadaşları PSO-KM-FCM metodunu önermişlerdir. Movielens 100K veri seti kullanılarak yapılan çalışmaların kök ortalama karesel hata performans değeri ve önerdiğimiz sistemin kök ortalama karesel hata değeri karşılaştırılmaları **şekil 4.22**'de verilmiştir.



Şekil 4.22 Movielens 100K veri seti ile elde edilen en iyi kök ortalama karesel hata değerleri

Siddiquee ve arkadaşları (2014) ve Sanandaj ve Alizadeh (2018) Movielens 1M veri setini kullanmıştır. Siddiquee ve arkadaşları öneri sisteminde bulanık mantık kullanırken, Sanandaj and Alizadeh işbirlikçi filtreleme ve içerik tabanlı filtreleme üzerine kurulu olan hybrid bir öneri sitemi önermişlerdir. Önerdikleri öneri sistemlerinin performansını kök

ortalama karesel hata performans değerlendirme kriteri ile değerlendirmişlerdir. Movielens 1M veri seti kullanılarak yapılan çalışmaların kök ortalama karesel hatası performans değeri ve önerdiğimiz sistemin kök ortalama karesel hata değeri karşılaştırılmaları **şekil 4.23**'te verilmiştir.



Şekil 4.23 Movielens 1M veri seti ile elde edilen en iyi kök ortalama karesel hatası değerleri

Bu çalışmada oy tahmini çok katmanlı algılayıcı, genelleştirilmiş ileri beslemeli ağ ve koaktif nöro bulanık çıkarım sistemi algoritmaları kullanılarak gerçekleştirilmiştir. Ardından, girdi değerlerine ve işleme elemanlarına genetik algoritma uygulanmıştır. Veri seti olarak Movielens 100K ve Movielens 1M veri setleri kullanılmıştır. **Şekil 4.22** ve **şekil 4.23**'te de gösterildiği üzere önerilen sistemin tahmin performansı birçok çalışmadan daha iyidir.

5. TARTIŞMA VE SONUÇ

Bu bölümde tez çalışması boyunca yapılan deneyler için genel bir değerlendirmeye ve gelecekte yapılması planlanan çalışmalara yer verilmiştir.

5.1 Değerlendirme

Bölüm 4.1.3'te, en iyi ortalama doğruluk ve ortalama ağırlıklı hatırlama değerlerinin 37692 adet doküman içeren veri seti ile elde edilen değerler dışında Tamamlayıcı Naif Bayes algoritmasına ait olduğu görülmüştür. Bu iki algoritmanın ortalama ağırlıklı hassasiyet değerleri birbirine çok yakındır. 75384 ve 150768 adet dokümandan oluşan veri setleri için aynıdır ve 1'e eşittir. 18846 adet dokümandan oluşan veri seti için Tamamlayıcı Naif Bayes performansı daha iyiyken, 37692 adet dokümandan oluşan veri seti için Naif Bayes algoritması daha iyidir. Genel olarak en iyi performans değerleri yumuşatma parametresi 0.1 seçildiğinde elde edilmiştir ve genel olarak lnorm normalizasyon parametresi bu iki algoritma için ve bütün veri setleri için en iyi performansı vermiştir. Tfidf ağırlık parametresinin her iki algoritmanın performansını iyileştirdiği görülmüştür. T test ve F test sonuçlarına göre iki algoritmanın performansları birbirine benzerdir.

Bölüm 4.1.4 sonuçları şu şekilde özetlenebilmektedir. Genel olarak en iyi ortalama eğitim süresi performansı Naif Bayes Algoritmasına aitken, en iyi ortalama doğru sınıflanan örnek yüzdesi performansı Tamamlayıcı Naif Bayes'e aittir. Naif Bayes algoritması ağırlıkları belirlerken sadece bir sınıfa ait olan örnekleri baz alarak belirleme yaptığından eğitim süresi daha azdır ve doğru sınıflanan örnek yüzdesi daha küçüktür. Tamamlayıcı Naif Bayes algoritması ise daha fazla örneği baz alarak ağırlık belirlemesi yaptığından ortalama eğitim süresi ve doğru sınıflanan örnek yüzdesi daha fazladır. Doğru sınıflanan örnek yüzdesi performansı doküman sayısı 37692 olan veri seti dışındaki tüm veri setleri için Tamamlayıcı Naif Bayes Algoritması için en iyi sonucu vermiştir. Naif Bayes algoritması kullanıldığında lnorm normalizasyon parametresi tüm veri setleri için en iyi ortalama doğru sınıflanan örnek yüzdesi oranını vermektedir. Tamamlayıcı Naif Bayes algoritması kullanıldığında 18846 adet dokümandan oluşan veri seti dışında tüm veri setleri için lnorm normalizasyon parametresi en iyi ortalama doğru sınıflanan örnek yüzdesini vermektedir.

Naif Bayes algoritması kullanıldığında 150768 adet dokümandan oluşan veri seti dışında tüm veri setleri için tfidf ağırlık parametresi en doğru sınıflanan örnek yüzdesini vermektedir.

Genel olarak bölüm 4.1'de Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının doğru sınıflanan örnek yüzdesi, ortalama eğitim ve test süreleri karşılaştırılması yapılmış, hangi algoritmanın hangi performans üzerinde daha iyi bir performans sergilediği gözlenmiştir ve en iyi performansları veren parametreler hakkında fikir verilmiştir.

Bölüm 4.2'de duygu analizi sonuçlarına yer verilmiştir. Doğruluk değeri eğilimlerinin tüm veri setleri ve her iki algoritma (Naif Bayes ve Tamamlayıc Naif Bayes) için aynı olduğu görülmektedir ve en iyi doğruluk değeri Tamamlayıcı Naif Bayes algoritması ile elde edilmiştir. Naif Bayes algoritması pozitif sınıfı dışında tüm sınıflar için en iyi F ölçüsü değerini vermektedir. Hatırlama değeri nötr ve negatif sınıfları için en iyi Naif Bayes algoritması ile sınıflandırılırken, en iyi hatırlama değeri pozitif sınıflaması Tamamlayıcı Naif Bayes algoritması ile yapılmıştır. Hassasiyet performans değerlendirme kriteri baz alındığında Tamamlayıcı Naif Bayes Algoritması negatif kelimeleri Naif Bayes algoritmasına göre daha iyi sınıflandırmaktadır. Verilen oylar ile duygular arasında yakın bir ilişki vardır.

Bölüm 4.3'te kullanıcı tabanlı bir öneri sistemi geliştirilmiş, kullanılan benzerlik hesaplama yöntemlerinin ve veri seti büyüklüğünün tahmin performansına olan etkisi gözlenmiştir. Veri seti büyüklüğünün artırılması ortalama mutlak hata ve kök ortalama karesel hatasını olumlu yönde etkilemiştir. 100,000 adet oydan oluşan veri seti kullanıldığında en iyi ortalama mutlak hata ve kök ortalama karesel hatası log bezerlik benzerlik hesaplama ölçütü ile elde edilirlen; en iyi hassasiyet, hatırlama ve F-ölçüsü Pearson ölçütü ile elde edilmiştir. 1 milyon adet oydan oluşan veri seti için ise en iyi ortalama mutlak hata ve kök ortalama karesel hatası merkezlenmemiş kosinüs ölçütü ile gözlenirken; en iyi hassasiyet, hatırlama ve F-ölçüsü Öklid ölçütü kullanıldığında gözlenmiştir.

Bölüm 4.4'te tahmin performansının yapay sinir ağları ve yapay sinir ağlarına genetik algoritma uygulanarak iyileştirilmesi amaçlanmıştır. Movielens 100K veri seti kullanıldığında en iyi performans çok katmanlı algılayıcı ve genetik algoritma bir arada kullanıldığında gözlenirken; Movielens 1M veri seti için genelleştirilmiş ileri besleme algoritması kullanıldığında gözlenmiştir. Bu çalışma sonucunda elde edilen sonuçların, diğer birçok çalışmanın sonucundan daha iyi olduğu görülmüştür.

Bu tez çalışmasında öneri sistemlerinde başarımı artırmak için yapay zeka tabanlı yaklaşımlar üzerinde çalışılmıştır. İlk önce Naif Bayes ve Tamamlayıcı Naif Bayes algoritmalarının performansı karşılaştırılmış, bu algoritmalara ait olan en iyi ortalama doğru sınıflanan örnek yüzdesi, ortalama kappa, ortalama doğruluk, ortalama güvenilirlik, ortalama ağırlıklı hassasiyet, ortalama ağırlıklı hatırlama ve ortalama ağırlıklı F1 derecesi değerlerini veren ağırlık, normalizasyon ve yumuşatma parametreleri elde edilmiştir. Tüm performans değerlendirme kriterleri baz alındığında genel olarak en iyi performanslar her iki algoritma için lnorm normalizasyon parametresi, tfidf ağırlık parametresi ve 0.1 yumuşatma parametresi kullanıldığında gözlenmiştir. Genel olarak en iyi ortalama eğitim süresi performansı Naif Bayes Algoritmasına aitken, en iyi ortalama doğru sınıflanan örnek yüzdesi performansı Tamamlayıcı Naif Bayes'e ait olmuştur. Ardından, her iki algoritma için en iyi doğruluk değerini veren parametreler kullanılarak duygu analizinde bulunulmuştur. Duygu analizinde ise en iyi doğruluk değerini Tamamlayıcı Naif Bayes algoritması göstermiştir. Daha sonra kullanıcı tabanlı bir öneri sistemi geliştirilmiş, kullanılan benzerlik hesaplama yöntemlerinin (Pearson, Öklid, Log Benzerlik, Merkezlenmemiş Kosinüs) öneri sistemi performansına olan etkisi gözlenmiştir. Son olarak, film oy tahmin performansı yapay sinir ağları (Çok Katmanlı Algılayıcı, Genelleştirilmiş İleri Besleme, Koaktif Nöro Bulanık Çıkarım Sistemi) ile iyileştirilmeye çalışılmış ve ardından yapay sinir ağlarına (girdi ve işleme elemanı) genetik algoritma uygulanmıştır. Çalışmalar başarıyla sonuçlanmış, bölüm 1.1'de listelenen amaçlara ve bölüm 1.2'de listelenen özgün değerlere ulaşılmıştır.

5.2 Gelecek Çalışmalar

Transformatörlerden çift yönlü enkoder gösterimler derin sinir ağı mimarisi kullanarak duygu analizinde bulunulması, öneri sistemi performasının derin öğrenme algoritmaları ile iyileştirilmesi, kullanılan veri seti sayısının artırılması ve elde edilen sonuçların tez çalışması sonucunda elde edilen çalışmalarla karşılaştırılması planlanmaktadır.

KAYNAKLAR

Adomavicius, G., Tuzhilin, A. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering 17(6), 734-749.

Agrawal, S., Jain, P. 2017. An Improved Approach for Movie Recommendation System. International Conference on IoT in Social, Mobile, Analytics and Cloud, 10-11 February, 336-342, Palladam, India.

Alhijawi, B., Kilani, Y. 2016. Using Genetic Algorithms for Measuring the Similarity Values between Users in Collaborative Filtering Recommender Systems. 15th International Conference on Computer and Information Science (ICIS), 26-29 June, 1-6, Okayama, Japan.

Almaghrabi, M., Chetty, G. 2018. A deep learning based collaborative neural network framework for recommender system. International Conference on Machine Learning and Data Engineering, 3-7 December, 121-127, Sydney, Australia.

Anagaw, A., Chang, Y. L. 2018. A new complement naive bayesian approach for biomedical data classification. Journal of Ambient Intelligence and Humanized Computing, Vol 10, 889–3897.

Anonymous. 2008. Web Sitesi: http://qwone.com/~jason/20Newsgroups/, Erişim Tarihi: 04.08.2017.

Anonymous. 2013. Web Sitesi: https://grouplens.org/datasets/movielens/, Erişim Tarihi: 25.08.2018.

Anonymous. 2015a. Web Sitesi: https://snap.stanford.edu/data/web-Movies.html, Erişim Tarihi: 23.08.2018.

Anonymous. 2015b. Web Sitesi: http://technobium.com/sentiment-analysis-using-mahout-naive-bayes/, Erişim Tarihi: 23.08.2018.

Anonymous. 2015c. Web Sitesi: https://optimization.mccormick.northwestern.edu/index.php/Conjugate_gradient_methods, Erişim Tarihi: 19.12.2019.

Anonymous. 2017a. Web Sitesi: https://www.sas.com/resources/asset/five-big-data-challenges-article.pdf, Erişim Tarihi: 19.08.2016.

Anonymous. 2017b. Web Sitesi: https://mahout.apache.org/docs/0.13.0/api/docs/mahout-mr/org/apache/mahout/cf/taste/impl/similarity/UncenteredCosineSimilarity.html, Erişim Tarihi: 25.07.2019.

Anonymous. 2019a Web Sitesi: http://scikit-learn.org/stable/modules/naif_bayes.html, Erişim Tarihi: 25.08.2018.

Anonymous. 2020a. Web Sitesi: https://mahout.apache.org/users/basics/algorithms.html, Erişim Tarihi: 12.12.2018.

Anonymous. 2020b. Web Sitesi: https://hpi.de/fileadmin/user_upload/fachgebiete/plattner/teaching/NaturalLanguageProcessing/NLP2016/NLP09_SentimentAnalysis.pdf, Erişim Tarihi: 01.09.2018

Anonymous. 2020c. Web Sitesi: https://www.mathworks.com/help/gads/what-is-the-genetic-algorithm.html, Erişim Tarihi: 04.09.2019.

Anonymous. 2020d. Web Sitesi: https://cedar.buffalo.edu/~srihari/CSE676/8.3%20BasicOptimizn.pdf, Erişim Tarihi: 19.12.2019.

Ashari, A., Paryudi, I., Tjoa, A. M. 2013. Performance comparison between naive bayes, decision tree and k-nearest neighbor in searching alternative design in an energy simulation tool, (IJACSA). International Journal of Advanced Computer Science and Applications, 4(11), 33-39.

Bagnall, A. J., Janacek, G. J.,Powell, M. 2005. A Likelihood Ratio Distance Measure for the Similarity Between the Fourier Transform of Time Series. 9th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining, 18-20 May, 737-743, Hanoi, Vietnam.

Barbosa, M. A. S., Jr, M. M. G. 2012. Access Point Design with a Genetic Algorithm. Sixth International Conference on Genetic and Evolutionary Computing, 25-28 August, 119-123, Kitakushu, Japan.

Bibault, J.E., Giraud, P., Burgun, A. 2016. Big Data and machine learning in radiation oncology: State of the art and future prospects. Cancer Letters, 382(1), 110–117.

Birtolo, C., Ronca, D., Armenise, R. 2011. Improving accuracy of recommendation system by means of Item-based Fuzzy Clustering Collaborative Filtering. 11th InternationalConference on Intelligent Systems Design and Applications, 22-24 November, 100-106, Cordoba, Spain.

Bostanci, E., Ar, Y. 2016. A genetic algorithm solution to the collaborative filtering problem. Expert Systems with Applications Vol 61, 122-128.

Burke, R. 2002. Hybrid Recommender Systems: Survey and Experiments. User Modeling and User-Adapted Interaction, 12(4), 331–370.

Chakrabarti, N., Das, S. 2019. Neural Networks and Collaborative Filtering. IEEE International Conference on System, Computation, Automation and Networking, 1-4, Pondicherry, India.

Chen, J., Zhao, C., Uliji, Chen, L. 2019. Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering. Complex & Intelligent Systems, Vol 6, 147-156.

Chen, M., Hao, Y., Hwang, K., Wang, L., Wang, L. 2017. Disease prediction by machine learning over big data from healthcare communities. IEEE Access, Vol 5, 8869-8879.

Chiroma, H., Abdulkareem, S., Abubakar, A., Zeki, A., Gital, A. Y., Usman, M. J. 2013. Co – Active Neuro-Fuzzy Inference Systems Model for Predicting Crude Oil Price Based on Organization for Economic Co-Operation and Development Inventories. 3rd International Conference on Research and Innovation in Information Systems, 27-28 November, 232-235, Kuala Lumpur, Malaysia.

Ciaramella, A., Cimino, M. G. C. A., Lazzerini, B., Marcelloni, F. 2010. Using Context History to Personalize a Resource Recommender via a Genetic Algorithm. 10th International Conference on Intelligent Systems Design and Applications, 29 November-1 December, 965-970, Cairo, Egypt.

Codina, V., Mena, J., Oliva, L. 2015. Context-Aware User Modeling Strategies for Journey Plan Recommendation, In: Lecture Notes in Computer Science, Ricci, F., Bontcheva, K., Conlan, O., Lawless, S. (eds), Vol 9146, Springer, Cham.

Da'u, A., Salim, N. 2019. Sentiment-aware deep recommender system with neural attention networks. IEEE Access, 2019; Vol 7, 45472-45484.

Devi, M. K. K., Samy, R. T., Kumar, S. V., Venkatesh, P. 2010. Probabilistic Neural Network Approach to Alleviate Sparsity and Cold Start Problems in Collaborative Recommender Systems. IEEE International Conference on Computational Intelligence and Computing Research, 28-29 December, 1-4, Coimbatore, India.

Devika, M D., Sunitha, C., Ganesh, A. 2016. Sentiment Analysis:A Comparative Study On Different Approaches. Procedia Computer Science, Vol 87, 44 – 49.

Dey, A. K. 2001. Understanding and using context. Personal and Ubiquitous Computing, Vol 5, 4-7.

Fan, X., Li, X., Du, F., Li, X.,Wei, M. 2016. Apply word vectors for sentiment analysis of APP reviews. 3rd International Conference on Systems and Informatics (ICSAI), 19-21 November, 1062-1066, Shanghai, China.

Friedman, N., Goldszmidt, M. 1996. Building Classifiers using Bayesian Networks, In: AAAI-96 Proceedings. 1277-1284.

Gediminas, A., Tuzhilin, A. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734-749.

Ghazanfar, M. A., Prugel-Bennett, A. 2011. Fulfilling the Needs of Gray-Sheep Users in Recommender Systems, A Clustering Solution. International Conference on Information Systems and Computational Intelligence, 18-20 January, China.

Gibson, G.J., Siu, S., Cowan, C. F. N. 1989. Application of multilayer perceptrons as adaptive channel equalizer. Adaptive Systems in Control and Signal Processing, Vol 23, 573-578.

Gong, S. J. 2008. The Collaborative Filtering Recommendation Based on Similar-Priority and Fuzzy Clustering. Workshop on Power Electronics and Intelligent Transportation System, 2-3 August, 248-251, Guangzhou, China.

Gong, S. J., Ye, H. W. 2009. An Item Based Collaborative Filtering Using Back Propagation Neural Networks Prediction. International Conference on Industrial and Information Systems, 24-25 April, 146-148, Haikou, China.

Goyal, A., Mehta, R. 2012. Performance comparison of naive bayes and J48 classification algorithms. International Journal of Applied Engineering Research, Vol 7, 1389-1393.

Guha, D. R., Patra, S. K. 2010. Cochannel interference minimization using wilcoxon multilayer perceptron neural network. International Conference on Recent Trends in Information, Telecommunication and Computing, 12-13 March, 145-149, Kochi, Kerala, India.

Gunawardana, A., Shani, G. 2009. A survey of accuracy evaluation metrics of recommendation tasks. Journal of Machine Learning Research, Vol 10, 2935–2962.

Guo-xia, W., He-ping, L. 2012. Summary of personalized recommendation system. Journal of Computer Engineering and Applications, Vol 3, 66-76.

Gupta, S., Lakra, S., Kaur, M. 2019. Sentiment Analysis using Partial Textual Entailment. International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), 14-16 February, 51-55, Faridabad, India.

Hamada, M., Abdulsalam, L.O., Hassan, M. 2018. Adaptive Genetic Algorithm for Improving Prediction Accuracy of a Multicriteria Recommender System. IEEE 12 th International Symposium on Embedded Multicore/Many-core Systemson-Chip, 12-14 September, 79-86, Hanoi, Vietnam.

Hassan, M., Hamada, M. 2017. Improving Prediction Accuracy of Multi-Criteria Recommender Systems Using Adaptive Genetic Algorithms. Intelligent Systems Conference, 7-8 Sept, 326-330, London, UK.

Hemachandra, S., Satyanarayana, R. V. S. 2013. Co-active neuro-fuzzy inference system for prediction of electric load. International Journal of Electrical and Electronics Engineering Research, Vol 3, 217-222.

Hinton, G. 2010. A practical guide to training restricted Boltzmann machines. Toronto, Canada: UTML TR 2010–003.

Hirakawa, G., Sato, G., Hisazumi, K., Shibata, Y. 2015. Data Gathering System for Recommender System in Tourism. 18th International Conference on Network-Based Information Systems, 2-4 September, 521-525, Taipei, Taiwan.

Hosseinpourpia, M., Oskoei, M. A. 2017. Genetic Algorithm Based Parameter Estimation for Multi-Faceted Trust Model of Recommender Systems. 5th Iranian Joint Congress on Fuzzy and Intelligent Systems, 7-9 March, 160-165, Qazvin, Iran.

Hu, M., Liu., B. 2004. Mining and Summarizing Customer Reviews. International Conference on Knowledge Discovery and Data Mining (KDD-2004), 22-25 August, Seattle, Washington, USA.

Huang, T., Lan, L., Fang, X., An, P., Min, J., Wang, F. 2015. Promises and challenges of big data computing in health sciences. Big Data Research, 2(1), 2–11.

Ilapakurti, A., Kedari, S., Vuppalapati, C. 2016. The Role of Big Data in Creating Sense EHR, an Integrated Approach to Create Next Generation Mobile Sensor and Wearable Data Driven Electronic Health Record (EHR), IEEE Second International Conference on Big Data Computing Service and Applications, 29 March-1 April, 293-296, Oxford, UK.

Inoue. M., Takenouchi, H., Tokumaru, M. 2016. Music Recommendation System Improvement Using Distributed Genetic Algorithm. Joint 8th International Conference on Soft Computing and Intelligent Systems and 17 th International Symposium on Advanced Intelligent Systems, 25-28 Aug, 627-630, Sapporo, Japan.

Inzunza, S., Juárez-Ramírez, R. 2016. Building Context-Aware Recommendation Systems: a Software Engineering Point of View. 4th International Conference in Software Engineering Research and Innovation, 27-29 April, 175-184, Puebla, Mexico.

Iqbal, U., Hsu, C. K., Nguyen, P. A., Clinciu, D. L., Lu, R., Syed-Abdul, S., Yang, H.C., Wang, Y. C., Huang, C. Y., Huang, C. W., Chang, Y. C., Hsu, M. H., Jian, W. S., Li, Y. C. 2016. Cancer-disease associations: A visualization andanimation through medical big data. Computer Methods and Programs in BioMedicine, 127(1), 44-51.

Isard, M., Yu, Y. 2009. Distributed Data-Parallel Computing Using a Highlevel Programming Language. SIGMOD Conference, 987-994.

Jadhav, S., Nalbalwar, S., Ghatol, A. 2012. Performance evaluation of generalized feedforward neural network based ECG arrhythmia classifier. IJCSI International Journal of Computer Science Issues, 9(4), 379-384.

Jain, S., Grover, A., Thakur, P. S., Choudhary, S. K. 2015. Trends, Problems and Solutions of Recommender System. International Conference on Computing, Communication and Automation, 15-16 May, 955-958, Noida, India.

Janjarassuk, U., Puengrusme, S. 2019. Product Recommendation Based on Genetic Algorithm. 5th International Conference on Engineering, Applied Sciences and Technology, 2-5 July, 1-4, Luang Prabang, Laos.

Jannach, D., Zanker, M., Felfernig, A., Friedrich, G. 2010. Recommender systems: an introduction.

Jayasena, K.P.N., Li, L., Xie, Q. 2017. Multi-modal multimedia big data analyzing architecture and resource allocation on cloud platform. Neurocomputing Vol 253, 135–143.

Jeon, T., Cho J., Lee, S., Baek, G., Kim, S. 2009. A Movie Rating Prediction System of User Propensity Analysis based on Collaborative Filtering and Fuzzy System. IEEE International Conference on Fuzzy Systems, 20-24 August, 507-511, Jeju Island, South Korea.

Jiang, J., Lu, J., Zhang, G., Long, G. 2011. Scaling-up Item-based Collaborative Filtering Recommendation Algorithm based on Hadoop. 2011 IEEE World Congress on Services, 4-9 July, 490-497, Washington, USA.

Jiang, L., Cai, Z., Zhang, H., Wang, D. 2013. Naif Bayes text classifiers: a locally weighted learning approach. Journal of Experimental & Theoretical Artificial Intelligence, 25(2), 273-286.

Jones, M. T. 2013. Introduction to approaches and algorithms of recommender system. IBM Corporation 2013, 1-7.

Katarya, R., Verma, O. P. 2017. An effective collaborative movie recommender system with cuckoo search. Egyptian Informatics Journal, Vol 18, 105-112.

Kaushik, C., Mishra, A. 2014. A scalable, lexion based technique for sentiment analysis. International Journal in Foundations of Computer Science & Technology (IJFCST), 4(5), 35-43.

Keiser, V., Dietterich, T. G. 2009. Evaluating Online Text Classification Algorithms for Email Prediction in TaskTracer, CEAS, Sixth Conference on Email and Anti-Spam, July 16-17 California, USA.

Kibriya, A., Frank, E., Pfahringer, B., Holmes, G. 2004. Multinomial naif bayes for text categorization revisited. Advances in Artificial Intelligence, Vol 3339, 488-499.

Kim, H. T., Kim, E., Lee, J. H., Ahn, C. W. 2010. A Recommender System Based on Genetic Algorithm for Music Data. 2nd International Conference on Computer Engineering and Technology, 16-18 April, 414-417, Chengdu, China.

Komiya, K., Sato, N., Fujimoto, K., Kotani, Y. 2011, Negation Naive Bayes for Categorization of Product Pages on the Web, Conference of Recent Advances in Natural Language Processing, RANLP, 12-14 September, 586-591, Hissar, Bulgaria.

Kuhn, M., Johnson, J. 2019. Feature Engineering and Selection: A Practical Approach for Predictive Models. CRC Press.

Kuncheva, LI. 2013. A Bound on Kappa-Error Diagrams for Analysis of Classifier Ensembles. IEEE Transactions on Knowledge and Data Engineering, Vol 25, 494-501.

Kwak, S. I., Choe, G., Kim, I. S., Jo, G. H., Hwang, C. J. 2015. A study of an modeling method of takagi–sugeno fuzzy system based on moving fuzzy reasoning and its application. CoRR, Vol 1511.02432, 1-24.

Landset, S., Khoshgoftaar, T.M, Richter, A.N. ve Hasanin T. 2015. A survey of open source tools for machine learning with big data in the Hadoop ecosystem. Journal of Big Data 2(24), 1-36.

Li, Q., Zhou, M. 2003. Research and design of an efficient collaborative filtering predication algorithm. Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, 29-29 August, 171-174, Chengdu, China.

Lin, K., Wang, J., Wang, M. 2014. A Hybrid Recommendation Algorithm Based on Hadoop. The 9th International Conference on Computer Science & Education (ICCSE 2014), 22-24 August, 540-543, Vancouver, BC, Canada.

Liu, G., Wu, X. 2019. Using Collaborative Filtering Algorithms Combined with Doc2Vec for Movie Recommendation. IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference, 15-17 March, 1461-1464, Chengdu, China.

Mai, J., Fan, Y., Shen, Y. 2009. A Neural Networks-Based Clustering Collaborative Filtering Algorithm in Electronic-Commerce Recommendation System. International Conference on Web Information Systems and Mining, 7-8 November, 616-619, Shanghai, China.

Maulik, U., Bandyopadhyay, S. 2000. Genetic algorithm-based clustering technique. Pattern Recognition, 33(9), 1455-1465.

Medhat, W., Hassan, A., Korashy, H. 2014. Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, Vol 5, 1093–1113.

Meyer, M., Pfeiffer, G. 1993. Multilayer perceptron based decision feedback equalizers for channels with intersymbol interference. IEEE Transactions Vol 140, 420-424.

Mohapatra, P., Mohapatra R. K., Sandhibigraha, B. 2019. Movie recommender system using improvised cuckoo search. International Journal of Innovative Technology and Exploring Engineering, Vol 8, 2869-2873.

Mukherjee, D., Banerjee, S., Bhattacharya, S., Misra, P. 2011. A Context-Aware Recommendation System Considering Both User Preferences and Learned Behavior. 7th International Conference on IT in Asia, 12-13 July, 1-7, Kuching, Sarawak, Malaysia.

Mussa, H. Y., D. Marcus, Mitchell, J. B. O., Glen, R. C. 2015. Verifying the fully "Laplacianised" posterior Naïve Bayesian approach and more. Journal of Cheminformatics, 7(27), 1-11.

Nithya, R., Ramyachitra, D., Manikandan, P. 2015. An efficient bayes classifiers algorithm on 10-fold cross validation for heart disease dataset. International Journal of Computational Intelligence and Informatics, 5(3), 229-235.

Panigrahi, S., Lenka, R. K., Stitipragyan, A. 2016. A hybrid distributed collaborative filtering recommender engine using apache spark. Procedia Computer Science, Vol 83, 1000-1006.

Papadakis, H., Panagiotakis, C., Fragopoulou, P. 2017. A synthetic coordinate based recommender system. Expert Systems with Applications, Vol 79, 8-19.

Paradarami, T. K., Bastian, N. D., Wightman, J. L. 2017. A hybrid recommender system using artificial neural networks, Expert Systems With Applications, Vol 83, 300–313.

Parvin, H., Moradi, P., Esmaeili S. 2018. A Collaborative Filtering Method Based on Genetic Algorithm and Trust Statements. 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), 28 Feb.-2 March, 13-16, Kerman, Iran.

Pasupathi, C., Kalavakonda, V. 2016. Evidence Based Health Care System Using Big Data for Disease Diagnosis, International Conference on Advances in Electrical,

Electronics, Information, Communication and Bio-Informatics (AEEICB16), 27-28 February, 743-747, Chennai, India.

Peerzade, S. S. 2017. Web Service Recommendation Using Pearson Correlation Coefficient Based Collaborative Filtering. International Conference on Energy, Communication, Data Analytics and Soft Computing, 1-2 August, 2920-2924, Chennai, India.

Prabowo, R., Thelwall, M. 2009. Sentiment analysis: A combined approach. Journal of Informetrics, 3(2), 143-157.

Pradeep, I. K., Bhaskar, M. J. 2018. Comparative analysis of recommender systems and its enhancements. International Journal of Engineering & Technology, Vol 7, 304-310.

Priyankka, R. P. J., Arivalagan, S., Sudhakar, P. 2019. Performance analysis of effective optimization algorithm based collaborative movie recommender systems. International Journal of Scientific Research and Review, Vol 8, 511-527.

Rahman, M. N., Esmailpour, A., Zhao, J. 2016. Machine Learning with Big Data An Efficient Electricity Generation Forecasting System. Big Data Research, 5(1), 9–15.

Rennie, J. D. M., Shih, L., Teevan, J., Karger, D. R. 2003. Tackling the Poor Assumptions of Naive Bayes Text Classifiers. Twentieth International Conference on Machine Learning (ICML-2003), Washington DC.

Resnick, P., Varian, H. R. 1997. Recommender systems. Communications of the ACM, 40(3), 56-58.

Ricci. F., Rokach, L., Shapira, B. 2011. Introduction to Recommender Systems handbook, In: Recommender Systems Handbook, Ricci. F., Rokach, L., Shapira, B., Kantor, P.B. (eds), Boston, US: Springer, 1-35.

Riyaz, P. A., Varghese, S. M. 2016. A Scalable Product Recommendations using Collaborative Filtering in Hadoop for Bigdata. Procedia Technology, Vol 24, 1393 – 1399.

Roy, D., Kundu, A. 2013. Design of movie recommendation system by means of collaborative filtering. International Journal of Emerging Technology and Advanced Engineering, 3(4), 67-72.

Sachdeva, K., Kaur, J., Singh, G. 2017. Image Processing on MultiNode Hadoop Cluster. International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), 15-16 December, 21-26, Mysuru, India.

Saito, T., Rehmsmeier, M. 2015. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE, 10(3),1-21.

Sanandaj, D. D., Alizadeh, S. H. 2018. A Hybrid Recommender System Using Multi Layer Perceptron Neural Network. 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium, 10-10 April, 7-13, Qazvin, Iran.

Sarwar, B., Karypis, G., Konstan J., Riedl, J. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. 10th International Conference on World Wide Web, 285-295.

Sayuti M, Sofyan, D. K. 2018. Manufacturing optimization using tsukamoto fuzzy inference system method: A case study in block paving and solid concrete block industry. 2nd International Mechanical and Industrial Engineering Conference, 2-9, Malang, Indonesia.

Shivhare, H., Gupta, A., Sharma, S. 2015. Recommender system using fuzzy c-means clustering and genetic algorithm based weighted similarity measure. IEEE International Conference on Computer, Communication and Control (IC4), 10-12 September, 1-8, Indore, India.

Shou-qiang, L., Ming, Q., Qing-Zhen, X. 2016. Research and Design of Hybrid Collaborative Filtering Algorithm Scalability Reform Based on Genetic Algorithm Optimization. 6th International Conference on Digital Home, 2-4 December, 175-179, Guangzhou, China.

Siddiquee, M. M. R., Haider, N., Rahman, R. M. 2014. A Fuzzy Based Recommendation System with Collaborative Filtering. The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), 18-20 December, 1-8, Dhaka, Bangladesh.

Smeureanu, I., Bucur, C. 2012. Applying supervised opinion mining techniques on online user reviews. Informatica Economică, 16(2), 81-91.

Soliman, T. H. A., El-Moemen Mohamed, S. A., Sewisy A.A. 2015. Developing a Mobile Location-Based Collaborative Recommender System for Geographic Information

Systems Applications. Tenth International Conference on Computer Engineering & Systems, 23-24 December, 267-273, Cairo, Egypt.

Speier, C., Valacich, J. S., Vessey, I. 1999. The Influence of Task Interruption on Individual Decision Making: An Information Overload Perspective. Decision Sciences, 30(2), 337-360.

Su, X., Khoshgoftaar, T. M. 2009. A Survey of Collaborative Filtering Techniques. Advances in Artificial Intelligence, Vol 2009, 1-19.

Subramaniyaswamy, V., Vijayakumar, V., Logesh, R., Indragandhi, V. 2015. Unstructured data analysis on big data using map reduce. Procedia Computer Science, Vol 50, 456 – 465.

Tan, S., Zhang, J. 2008. An empirical study of sentiment analysis for chinese documents. Expert Systems with Applications, 34(4), 2622-2629.

Verma, J. P., Patel, B., Patel, A. 2015. Big Data Analysis: Recommendation System with Hadoop Framework. IEEE International Conference on Computational Intelligence & Communication Technology, 13-14 February, 92-97, Ghaziabad, India.

Verma, S. K., Mittal, N., Agarwal, B. 2013. Hybrid Recommender System based on Fuzzy Clustering and Collaborative Filtering. 4th International Conference on Computer and Communication Technology (ICCCT), 20-22 September, 116-120, Allahabad, India.

Wang, C., Zheng, Z., Yang, Z. 2014. The Research of Recommendation System Based on Hadoop Cloud Platform. The 9th International Conference on Computer Science & Education (ICCSE 2014), 22-24 August, 193-196, Vancouver, Canada.

Wang, H., Xu, Z., Fujita, H., Liu, S. 2016. Towards felicitous decision making: An overview on challenges and trends of Big Data. Information Sciences. 367–368, 747–765.

Wang, L., Niu, J., Yu, S. 2020. SentiDiff: Combining Textual Information and Sentiment Diffusion Patterns for Twitter Sentiment Analysis. IEEE Transactions on Knowledge and Data Engineering, 32(10), 2026-2039.

Woldemariam, Y. 2016. Sentiment analysis in a cross-media analysis framework. IEEE International Conference on Big Data Analysis (ICBDA), 12-14 March, 1-5, Hangzhou, China.

Xia, R., Zong, C., Li, S. 2011. Ensemble of feature sets and classification algorithms for sentiment classification. Information Sciences, 181(6), 1138-1152.

Xian, L. 2010. Artificial Intelligence and Modem Sports Education Technology. International Conference on Artificial Intelligence and Education (ICAIE), 29-30 October, 772-776, Hangzhou, China.

Xiang, B., Zhang, Z., Dong, H., Wu, Q., Hu, L. 2013. Research of Mobile Recommendation System Based on Hybrid Recommendation Technology. 3rd International Conference on Consumer Electronics, Communications and Networks, 20-22 November, 508-512, Xianning, China.

Ye, Q., Zhang, Z., Law, R. 2009. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Systems with Applications, 36(3), 6527-6535.

Yera, R., Martinez, L. 2017. Fuzzy Tools in Recommender Systems: A Survey. International Journal of Computational Intelligence Systems, Vol 10, 776–803.

Yu, H. F., Hsieh, C. J., Si, S., Dhillon, I. 2012. Scalable Coordinate Descent Approaches to Parallel Matrix Factorization for Recommender Systems. IEEE 12th International Conference on Data Mining, 10-13 December, 765-774, Brussels, Belgium.

Zhang, Z., Ye, Q., Zhang, Z., Li., Y. 2011. Sentiment classification of Internet restaurant reviews written in Cantonese. Expert Systems with Applications, 38(6), 7674-7682.

Zhao, Z. D., Shang, M. S. 2010. User-Based Collaborative-Filtering Recommendation Algorithms on Hadoop. Third International Conference on Knowledge Discovery and Data Mining, 9-10 January, 478-481, Phuket, Thailand.

Zheng, S. 2014. Naive Bayes Classifier: A Mapreduce Approach., Master of Science, Graduate Faculty of the North Dakota State University of Agriculture and Applied Science, 47, Fargo, North Dakota.

Zhou, L., Pan, S., Wang, J., Vasilakos, A.V. 2017. Machine learning on big data: Opportunities and challenges. Neurocomputing, Vol 237, 350–361.

Zhou, W., Wen, J., Qu, Q., Zeng, J., Cheng, T. 2018. Shilling attack detection for recommender systems based on credibility of group users and rating time series. PLoSONE, 13(5), 1-17.

Zhou, Y., Wilkinson, D., Schreiber, R., Pan, R. 2008. Large-Scale Parallel Collaborative Filtering for the Netflix Prize. International Conference on Algorithmic Applications in Management, Vol 5034, 337–348, Berlin, Heidelberg.